



BRITISH CRIME SURVEY USER GUIDE

2006/07

**CRIME SURVEYS PROGRAMME
RESEARCH, DEVELOPMENT AND STATISTICS
CRIME REDUCTION AND COMMUNITY SAFETY GROUP
HOME OFFICE**

CONTENTS

CHAPTER 1: BACKGROUND TO THE BCS	1
CHAPTER 2: SAMPLE DESIGN AND METHODOLOGY	2
FIELDWORK	2
WEIGHTING	3
QUESTIONNAIRE	3
OFFENCE CODING.....	4
CHAPTER 3: BCS ANALYSIS.....	6
BCS MAIN MODULES.....	6
INDIVIDUAL BASED ANALYSIS	6
HOUSEHOLD BASED ANALYSIS	6
INCIDENT BASED ANALYSIS	6
RATES BASED ANALYSIS	6
BCS DRUGS MODULE	7
CHAPTER 4: BCS DATA FILES	8
CASE IDENTIFICATION	8
WHAT CASES TO INCLUDE	8
ROUTING.....	8
RECALL PERIOD	8
INCIDENTS OUTSIDE ENGLAND & WALES	8
OFFENCE CODES	8
VARIABLE NAMES.....	8
MULTIPLE RESPONSE VARIABLES.....	9
DERIVED VARIABLES	9
DON'T KNOWS AND REFUSALS	9
BASE NUMBER	9
SIGNIFICANCE TESTS	9
CHAPTER 5: BCS DRUGS MODULE.....	11
YOUNG ADULT BOOST	11
DRUGS MODULE OVER TIME	11
MISSING DATA ON DRUGS MODULE	12
NAMES OF DRUGS.....	12
CHAPTER 6: METHODOLOGICAL LIMITATIONS.....	14
SAMPLING ERROR.....	14
NON-RESPONSE	14
RECALL	14
UNWILLINGNESS TO REPORT	14
DEFINITIONS OF CRIME.....	15
CHAPTER 7: BCS DATASETS.....	16
ACCESS TO BCS DATA.....	16
CHANGES FROM 2006-07	16
CHAPTER 8: BCS PUBLICATIONS.....	18

BCS PUBLICATIONS	18
FORTHCOMING PUBLICATIONS	18
APPENDIX A	19
APPENDIX B	20
APPENDIX C	22
APPENDIX D	32
APPENDIX E.....	73

CHAPTER 1 – Background to the BCS

The British Crime Survey (BCS) is a well-established study and one of the largest social research surveys conducted in England and Wales. The BCS asks adults, aged 16 and over, in private households about their experiences of criminal victimisation in the previous 12 months, regardless of whether or not they reported the incident to the police. Respondents are asked about their household's experiences of property crimes such as vehicle-related thefts and burglary, and their own experience of personal crimes such as assaults.

As well as providing estimates of victimisation, the BCS collects extensive information about victims of crime, the circumstances in which incidents occur and the behaviour of offenders in committing crimes. In this way, the survey provides information to inform crime reduction measures and to gauge their effectiveness. Although there have been changes to the design of the survey over time, the wording of the questions that are asked to elicit victimisation experiences have been held constant throughout the life of the BCS – therefore it is the best guide to long-term trends. The BCS also provides data on matters such as contacts between the public and the police, attitudes towards aspects of the criminal justice system and self-reported use of illicit drugs.

The survey was first conducted in 1982; with further cycles in 1984, 1988, 1992, 1994, 1996, 1998, 2000 and 2001. The first survey was carried out in England, Wales & Scotland (hence "British" Crime Survey), as was the third survey. The BCS now only covers England and Wales as Scotland now has its own survey, as does Northern Ireland. In 2001, the BCS moved to an annual format with continuous sampling.

As a survey of private households, the BCS does not cover commercial victimisation, e.g. thefts from businesses and shops, and frauds. The BCS crime count also excludes so-called victimless crimes (e.g. illegal drug use) and crimes against non-resident overseas visitors.

The booklet linked below, examines how the BCS has changed and what has happened to trends in crime and people's perceptions about crime over the last 25 years.

<http://www.homeoffice.gov.uk/rds/pdfs07/bcs25.pdf>

CHAPTER 2 – Sample design and methodology

The BCS sample is designed to give both a representative cross-section of private households in England and Wales and of individuals aged 16 and over living in them.

The survey does not cover the population resident in institutions such as halls of residence, those in residential care, those in prison, or members of the armed forces living on armed force bases. Nor does it currently cover the experiences of those aged under 16.

The BCS has grown significantly in response to local level analysis requirements. The core sample size has increased from around 11,000 in the earlier cycles to around 47,000 in 2006/07. In addition to this, the BCS interviews a boost sample of 2,000 young adults (16 to 24 year olds) who are identified through screening at the core addresses. Up until the 2007/08 BCS the survey incorporated a boost sample of non-white respondents. For more information about when the boost samples were included on the BCS see Appendix A.

Following a methodological review¹, significant changes were made to the design of the BCS in 2001. The two main changes introduced at this time were an increase in the sample size and a move to continuous fieldwork. The move to continuous data collection was also associated with a change in the survey reference period, with respondents being asked about events that had taken place in the last full 12 months from the date of interview. Prior to this respondents had been asked about events that had taken place in the last calendar year. For example, the 2000 BCS asked about experiences of crime from 1st January 1999 to the date of the interview.

In 2004/05 the sample size of the BCS was increased to approximately 46,000 interviews per year, with a minimum of 1,000 interviews in each of the 42 Police Force Areas (PFA), to provide more precise estimates at that level.

Since 1992 the Small Users Postcode Address File (PAF) has been used as the sampling frame for the BCS. The PAF, listing all postal delivery points in the country, represents the most complete register of household addresses. Each 'postcode sector' is used as a Primary Sampling Unit (PSU). Sampling is carried out to ensure that a nationally representative sample of sectors is issued each quarter.

In all rounds of the BCS, a relatively high response rate has been achieved (see Appendix A). The main reasons for non-response at eligible addresses are (a) refusal either by the selected person or by the household before the respondent has been selected and (b) non-contact.

A more detailed description of the sample design can be found in the technical reports, which are available for each survey year. (<http://www.homeoffice.gov.uk/rds/pdfs07/bcs0607tech1.pdf>).

FIELDWORK

All selected addresses are sent a letter and a leaflet from the Home Office in advance of an interviewer calling at the address. These explain a little about the survey, why their address has been selected, and tries to answer some questions that potential respondents may have, such as issues relating to confidentiality. On making contact at an address, an interviewer selects one individual aged 16 or over per household for interview by a random method.

Respondents are interviewed face-to-face in their own home. Since 1994, the BCS has used Computer Assisted Personal Interviewing (CAPI). Responses to questions are entered directly

¹ Lynn, P. and Elliot, D. (2000) The British Crime Survey: A review of methodology. London: National Centre for Social Research.

into the laptop by the interviewer. There is also a self-completion section on sensitive topics at the end of the main interview which is completed by CASI (Computer Assisted Self Interviewing).

A more detailed description of the fieldwork can be found in the technical reports, which are available for each survey year (<http://www.homeoffice.gov.uk/rds/pdfs07/bcs0607tech1.pdf>).

WEIGHTING

Weighting on the BCS serves three purposes: to correct for different sampling rates (this is called *design weighting*); to account for differential rates of response between different groups of people (the BCS uses *calibration weighting* to take account of non-response in the core sample) and to take account of 'series' of similar incidents.

First, the raw data are weighted to compensate for unequal probabilities of selection. These include: the individual's chance of participation being inversely proportional to the number of adults living in the household; the over-sampling of smaller police force areas and the selection of one household at multi-household addresses. More information on design weights can be found in the technical reports, which are available for each survey year (<http://www.homeoffice.gov.uk/rds/pdfs07/bcs0607tech1.pdf>).

Second, calibration weighting is used to adjust for differential response rates. A review of the BCS methodology recommended that the calibration method be adopted in the BCS (Lynn and Elliot, 2000²). The calibration weighting in the BCS is designed to make adjustment for known differentials in response rates between different subgroups (age by gender and region) and households with different age and gender composition. For example, a household containing a 24-year-old male living alone may be less likely to respond to the survey than a household containing a 24-year-old male living with a young partner and a child. The procedure therefore gives different weights to different household types based on their age/sex composition in such a way that the weighted distribution of individuals in the responding households matches the known distribution in the population as a whole.

The individual and household design weights are calibrated to produce the individual weight **indivwgt**, which is used for individual based analysis, and the household weight **hhdwgt**, which is used for household based analysis. For incident-based analysis, the weight **weighti** is used. For analysis confined to 16-24 year olds a weight based on 16-24 year olds from the main sample and those in the young adults boost sample should be used (**ypcwgt**).

QUESTIONNAIRE

The questionnaire has a complex structure, consisting of a set of core modules asked of the whole sample, a set of modules asked only of different sub-samples, and self-completion modules asked of all 16-59 year olds. Within some modules there is often further filtering so that some questions are only asked of even smaller sub-samples. The precise modules asked on the survey vary from year to year as do the exact modules asked of the core and boost samples. More information about the questionnaire structure and content can be found in the technical reports, which are available for each survey year (<http://www.homeoffice.gov.uk/rds/pdfs07/bcs0607tech1.pdf>).

There are two stages to the questionnaire for measuring experiences of victimisation. First, respondents are asked a series of screener questions on the main part of the questionnaire to assess if they have been a victim of crime. For each type of incident, respondents are asked how

² Lynn, P. and Elliot, D. (2000) The British Crime Survey: A review of methodology. London: National Centre for Social Research.

many times they experienced that type of incident in the reference period. If they experienced such an incident on more than one occasion during the period and they regard all the incidents as very similar, where the same thing was done under the same circumstances and probably by the same people, then the incidents are known as 'series' incidents.

The wording of the screener questions has been kept consistent since the BCS began to ensure comparability across the surveys. They are designed to ensure that all incidents of crime within scope of the BCS, including relatively minor ones, are mentioned.

Within the screener questions a crucial distinction exists between household incidents and personal incidents. All vehicle-related and property-related crimes are considered to be household incidents, and respondents are asked about whether anyone currently residing in the household has experienced any incidents within the reference period. Personal incidents refer to all crimes against the individual and only relate to things that have happened to the respondent personally, and not to other people in the household.

Second, those who have been victimised are asked detailed questions about exactly what happened. Since 1996 the BCS has recorded a maximum of six Victim Forms - three long Victim Forms and three short Victim Forms. This is to reduce the burden put on respondents. The three long Victim Forms collect full details of what happened in the incident. The three Short Victim Forms only collect the key information required to code an incident into an offence.

It is important to explain why we use the Victim Form count rather than the responses to the screener questions to produce the count of crime. First, the level of detail on the Victim Form allows an offence to be given a legal offence code following objective criteria. Some incidents from the screeners turn out not to have been criminal incidents at all when followed up in detail (between 7% and 10% are not crimes). Also the final offence code may not correspond to the screener from which the Victim Form arose. For example, an incident elicited from the burglary screener may turn out to be a case of vandalism. Second, an incident can be double counted on the screener. Despite careful wording of the questions the respondent may report a single incident on two screeners.

Since 1992 respondents aged 16-59 have been asked to complete self-completion questionnaires on drug use, drinking behaviour, stalking etc.

OFFENCE CODING

Based on the information collected on the victim form an offence code is assigned to each crime. This is done by coders employed by the survey contractor. Home Office researchers check a sample of offences to ensure accuracy of the coding. Coding reliability tests are carried out periodically to measure consistency between the coders employed by the survey contractor and Home Office researchers. The last coding reliability test was in 2004, and the next one is planned for 2008.

The 'Coding guidance' can be found in the technical reports, which are available for each survey year (<http://www.homeoffice.gov.uk/rds/pdfs07/bcs0607tech1.pdf>).

Note that although threats are valid BCS offence codes they are not included in most BCS analyses conducted by the Home Office as the incidents do not always meet the criteria of a criminal offence. Also sexual offences are not included in the count of total crime because the small number of incidents picked up by the survey means the results would be unreliable. (Instead, more reliable information collected on self-completion modules at the end of the questionnaire are used to calculate sexual victimisation rates separately (see chapter 6)).

In Home Office analysis it is the offence coded Victim Forms which are used to produce the count of crime. The use of the Victim Form to count crime has been criticised on two counts. The first is that the number of Victim Forms is limited to six per respondent and this places a limit on the number of crimes which can be counted using the Victim Forms. Whilst this is true, our analysis suggests that the impact on the count is minimal. The second criticism is that the Home Office caps series incidents at five in calculating Victim Form based offence rates. This was an analytical decision and designed to avoid the risk that a small proportion of cases with extreme values can produce unreliable trend estimates. This cap has been applied since the first BCS and has been retained to ensure a consistent methodology is applied, which is required for measuring trends over time.

CHAPTER 3 - BCS analysis

BCS MAIN MODULES

Individual-based analyses

Individual-based analysis is carried out when the intention is to make statements about the characteristics, attitudes or experiences of adults in the sample. It is important to recognise that statements about the characteristics of victims are not the same as incident-based analysis in which statements are made about incidents (see incident-based analysis). This is because a victim can experience more than one incident. ALL INDIVIDUAL-BASED ANALYSIS SHOULD BE WEIGHTED BY INDIVWGT (WEIGHTA PRIOR TO 1996 SURVEY).

Household-based analyses

Household based analysis is carried out when the intention is to make statements about the characteristics or experiences of households in the sample. Again, the characteristics of the households are not the same as incident-based analysis in which statements are made about incidents as a household can experience more than one incident. The most common type of household-based analysis is analysis in which statements are made about households who were victims of household crimes. ALL HOUSEHOLD-BASED ANALYSIS SHOULD BE WEIGHTED BY HDDWGT (WEIGHTB PRIOR TO 1996 SURVEY).

Incident-based analysis (victim form analysis)

This comprises analyses where the intention is to make statements about the nature of incidents of crime using the information collected on the Victim Form. Incident-based analysis usually examines the nature of specific offence types or compares different types of offence. ALL INCIDENT-BASED ANALYSIS SHOULD BE WEIGHTED BY WEIGHTI.

Rates-based analysis (incidence rates and prevalence rates)

The Home Office produces four BCS measures of the extent of crime in England and Wales:

- **Incident rates** give the number of crimes experienced per 10,000 households or adults. Rates of household offences are based on mean numbers of incidents per 10,000 households and personal offences are based on mean numbers of incidents per 10,000 adults. Incident rates can be grossed up by population figures to give the total number of crimes in England and Wales for a particular BCS year.
- **Prevalence rates** (also known as risks) give the percentage of households or adults in the sample who were the victim of an offence once or more during the recall period. Unlike incidence rates, prevalence rates take no account of the number of victimisations experienced. It simply divides people into victims and non victims and describes the 'risk' of being a victim of crime.
- **Crime count**
The total number of BCS crimes (calculated by applying the incident rates to population figures)
- **Numbers of victims**
The total number of victims (calculated by applying the prevalence rates to population figures)

Rates analysis is essentially a household or individual based analysis as the aim is to make statements about the number of incidents of crime per household/adult or the proportion of households/adults victimised.

Rates based analysis is the most complicated type of analysis carried out. The general logic is outlined below.

- The number of incidents each Victim Form refers to is calculated. In the case of single incident Victim Forms this is simply 1. For 'series' incidents this is the total number of incidents in the series.
- For 'series' incidents the number of incidents is capped at 5. Therefore if someone reports 10 incidents in a 'series' only 5 are counted (see chapter 2).
- The hierarchical data held on the Victim Form is then collapsed into a flattened form as follows. The number of incidents of different offence types experienced by the household or respondent are counted. For example, each household/respondent will have a count of burglary codes, a count of common assault codes, and so on. For each household/respondent, the number of incidents of the specified offence type across all their Victim Forms is summed. This gives the total number of incidents for each offence type experienced by that household or respondent.
- The rates can then be calculated on the basis of adults/households or incidents. For personal offences the weight variable 'indivwgt' is used and for household offences the weight variable 'hhdwgt' is used.

As well as looking at rates for the whole sample, it is possible to examine rates for different sub-samples. For example, in examining the risks of vehicle-related offences the sample is usually restricted to vehicle-owning households. Similarly rates of bicycle theft are often given just for bicycle-owning households. Moreover, in examining rates separately for households or individuals with different characteristics the BCS has shown how the risks of victimisation vary across the population.

The BCS is able to provide a robust and consistent measure of the general experience of violence in England & Wales. However, for more serious violence (such as attempted murder or robbery), their relatively rare occurrence means that insufficient numbers of survey respondents report being victims of such crimes. As a victim based survey, murders are not included on the BCS. It should also be noted that the Home Office do not usually present BCS results on sexual offences (rape, attempted rape and indecent assault) based on analysis from the victim form. This is because the number of such incidents counted on the victim forms is too small for reliable analysis. Instead, information collected on self-completion modules are used to calculate sexual victimisation.

BCS DRUGS MODULE

Since 1996 the BCS has included a comparable self completion module of questions on illicit drug use. BCS respondents aged 16 to 59 years old are eligible for the drugs module of the survey. Respondents complete the drugs module by themselves on a laptop computer, which allows them to feel more at ease when answering questions on illicit behaviour due to increased confidence in the privacy and confidentiality of the survey.

More information about the drugs module can be found in chapter 5.

CHAPTER 4 – BCS data files

The BCS is a large and complex data set which provides a rich source of data for analysis. However it is important that anyone undertaking analysis understands the structure of the data. Listed below are some general points about the data and how Home Office analysts use it.

CASE IDENTIFICATION

Each individual respondent has a unique case identifier (rowlabel) consisting of an eight digit number. This identifier is the same on each data file on which information is held about the respondent.

ROUTING

From the 1994 BCS onwards the questionnaires in the technical reports provide the routing instructions above the questions in square brackets. The routing instructions for surveys before 1994 are given above the questions and alongside responses to questions.

REFERENCE PERIOD

This is the time over which BCS respondents are asked to report offences they had experienced. Following the introduction of continuous sampling in 2001/02 the BCS reference period for offences relates to the 12 calendar months prior to interview. Prior to this the reference period was between 1 January and 31 December of the previous year.

The Victim Forms hold information on all incidents reported to the survey. Although respondents are asked to tell us about incidents that happened in the appropriate reference period, some incidents outside the reference period are reported. In all incident-based analysis, only incidents occurring in the correct reference period should be retained.

INCIDENTS OUTSIDE ENGLAND & WALES

Since 1992 incidents which occurred outside of England and Wales have been given a short victim form and a valid offence code. Prior to 1992, incidents outside of England & Wales were given an invalid offence code. For incident-based analysis only those incidents which occurred within England and Wales should be retained.

OFFENCE CODES

All incidents reported in the victim forms are assigned an offence code. However, not all incidents are included in the analyses conducted by Home Office researchers. Incidents that occur outside England & Wales are excluded, so are duplicate incidents and those committed by a policeman or woman on duty (The 'Coding guidance' can be found in the technical reports, which are available for each survey year (<http://www.homeoffice.gov.uk/rds/pdfs07/bcs0607tech1.pdf>)).

VARIABLE NAMES

BCS variable names can be found in the technical report alongside the question to which they refer.

In the 1984, 1988 and 1992 BCS, variable names are the question numbers prefixed by a letter³ to indicate which part of the questionnaire it refers to. So variables on the Main are prefixed M, those on the Victim Form V, those on the demographic D, and those on the Self-Completion SC. Variables on the 1982 data set have names. Further details are available in the technical reports for each BCS survey year (<http://www.homeoffice.gov.uk/rds/pdfs07/bcs0607tech1.pdf>)

³ Respondents are randomly assigned to one of four part-sample modules A, B, C or D. These have previously been referred to as 'follow-up' modules

MULTIPLE RESPONSE VARIABLES

Multiple response variables are those questions which permit respondents to choose more than one answer from the list of available options.

From the 2001 BCS onwards, multiple response variables are set up so a set of variables (equal to the full number of possible answers that could be given) hold the responses to the question. The first variable records whether or not the first option was selected; the second records whether or not the second option was selected, and so on. So for example, the variable Whemot has ten variables, WhemotA to WhemotL. If the respondent answers codes 2 and 4, WhemotB and WhemotD will be coded 1 and the remaining eight variables will be coded 0.

From the 1994 survey up to and including the 2000 survey, multiple response variables were coded differently: in the same way as above the same number of variables as there are possible answers are set up. However, the first variable recorded the first answer given; the second recorded the second answer given, and so on. So in the example above there would be ten variables called whemot00 to whemot09. If a respondent answered codes 2 and 4 in response to this question, whemot00 would hold the code '2' and whemot01 '4'. All the other variables would be system missing.

Prior to the 1994 BCS, multiple response variables are denoted by the 'mr' suffix.

DERIVED VARIABLES

In addition to the questions directly asked of respondents, the BCS data files also include derived variables. There are two sets of derived variables; those produced by the survey contractor (can be found in the latest technical report <http://www.homeoffice.gov.uk/rds/pdfs07/bcs0607tech1.pdf>) and those produced by Home Office analysts. From 2006/07, the derived variables produced by Home Office analysts have been included on the datasets so researchers can replicate the analysis produced in various publications. For more information on derived variables, see Appendices D and E.

DON'T KNOW & REFUSAL CODES

Respondents are not usually explicitly given the options 'don't know' or 'refusal'. However, for every question respondents may say they do not know or refuse to answer and these are valid responses. The code for don't know is '9' for code frames up to 7, '99' for code frames up to 97, and so on. The code for refusal is '8', '98' and so on.

Prior to the 1994 BCS the reverse holds. '9', '99' and so on denote refusals or question not completed in error; '8', '98' and so on denote don't knows.

In most Home Office analysis refusal codes are excluded. Don't know codes are also usually excluded unless there is interest in these responses, for example in the case of attitudinal questions.

BASE NUMBER

The number of cases upon which analysis is based is important as it influences the precision (standard error) of the estimates. The Home Office does not generally publish estimates based on less than 50 cases.

SIGNIFICANCE TESTS

Because BCS estimates are subject to sampling error, differences between estimates from successive years of the survey or between population subgroups may occur by chance. Tests of statistical significance are used to identify which differences are unlikely to have occurred by chance. The BCS has a complex design and therefore uses specialist software to calculate 'complex standard errors' for key BCS estimates.

For ad hoc analysis, the complex standard error for a variable can also be estimated using an estimation of the Deft. Deft is the ratio of the actual standard error for the complex design to the standard error from a Simple Random Sample of the equivalent size. The Home Office tends to use a Deft of 1.2 in BCS analysis (see Appendix B for table of design effects used in 2006/07 data).

In Home Office published reports, tests at the five per cent significance levels are applied (the level at which there is a one in 20 chance of incorrectly identifying a difference solely due to chance variation), unless otherwise stated.

CHAPTER 5 – BCS drugs module

Since 1996 the BCS has included a self completion module on illicit drug use. Data from the module is currently published in an annual statistical bulletin, with results based on 16-59 year olds who have completed the drugs module and 16 to 24 year olds interviewed as part of the BCS young adult boost.

YOUNG ADULT BOOST

Since 2001, the BCS has included a boost of approximately 2,000 interviews with 16 to 24 year olds identified through screening at the main address. The young adult questionnaire covers fewer topics than those asked in the main questionnaire and questions on household victimisation are also excluded as these are answered by the main respondent. The young adult boost file includes all young people (16-24) from both the main and boost samples in one dataset.

For the young adult dataset, only an individual weight is computed as the household weight is the same as that calculated for the main data file. The weight variable from 2006/07 onwards is called **ypcwgt**, prior to this the weight variable was called **indivyb**.

DRUGS MODULE OVER TIME

Since the introduction of the drug module in 1992 there have been relatively few changes made to the questions asked.

1996 module:

In 1996, respondents answered Yes/No to the following questions:

- Have you ever HEARD of X?

With the range of responses Yes/No/Don't want to answer, respondents were asked:

- Have you EVER taken X?
- Have you taken X in the LAST YEAR?
- Have you taken X in the LAST MONTH?

Respondents were routed out accordingly, for example they were only asked if they had taken X in the LAST YEAR, if they responded positively to having EVER taken (obviously the respondent would have said that they had HEARD of X).

Respondents were also asked: 'Have you injected or had someone else injected you with any drug not prescribed by a doctor?'

1998 module:

The 'inject' question that was introduced in 1996 was dropped.

2000 module:

Replica of 1998 module.

2001/02 module:

There were several changes made to the questions in this round.

- The 'ever HEARD' question was dropped and put in as a response in the 'EVER taken' question. So now when asking 'Have you EVER taken X?' the responses available were

Yes/No/Never heard of it/Don't want to answer (obviously not asked for the taken anything unknown questions).

- The 'taking anything else that you thought was a drug' changed to 'taking anything else that you knew or thought to be a drug'.

There was also two new sets of questions introduced asked only of 16-24 year olds (core & boost), concerning 'age when first used X?' and 'how easy to get hold of any illegal drugs/X?'. Responses for the ease of access questions were in the form of a Likert Scale – ranging from 'very easy' to 'impossible' (as well as usual refusal categories).

2002/03 module:

The questions on ease of access to illegal drugs were dropped after June 2002. In their place, a new set of questions asking about frequency of drug use were added for 16-24 year olds who had taken drugs in the last 12 months. Respondents were asked: 'How often during the last 12 months have you taken X?' with the responses ranging from 'Every day' to 'Once or twice a year'. The questions on frequency of use were asked in July – December 2002 only.

2003/04 module:

Replica of 2002/03 module with questions on ease of access to illegal drugs excluded and questions on frequency of use included. Also, the questions on age of first use were asked to all those aged 16 to 59, not just 16 to 24 year olds.

2004/05 module:

Questions on age of last use of each drug were added for all those aged 16 to 59 who had ever taken particular drugs.

2005/06 module:

Questions on Ketamine introduced and questions on age when first taken LSD/Acid/Magic Mushrooms/Methadone/Physeptone removed.

2006/07 module:

Questions on age of first use and last use of each drug were removed.

MISSING DATA ON DRUGS MODULE

Those people who refuse to take part in self-completion drug module are coded as missing in the dataset. The BCS drugs dataset also codes as missing those people who reported having taken Semeron. Analysis showed that those people, who said they had used this fake drug, tended to have also said they had taken all other drugs (yes to ever/year/month). As these people have given unreliable answers, they are coded as missing in the data set. There are only a handful of respondents who have claimed having used Semeron, so their exclusion makes minimal difference to the sample size.

NAMES OF DRUGS

The list of drugs asked about in the BCS has remained relatively consistent. However, over time the street names for drugs tend to change. To reflect this, the BCS has often had to add additional examples to each drug. This is necessary so as to maximise the capture of use.

The table on page 13 shows the drugs asked about in each round and the list of examples that is used to describe the drug.

There are also questions on:

- Taking pills/powder unknown.
- Smoking something unknown.
- Taking something else they thought (knew) was a drug.

Table 1: Street name of drugs included in the BCS drugs module 1996-2006/07

Drug	1996	1998	2000	2001/02	2002/03	2003/04	2004/05	2005/06	2006/07
Amphetamines	Speed Whizz Uppers	Speed Whizz Uppers	Speed Whizz Uppers Billy	Speed Whizz Uppers Billy	Speed Whizz Uppers Billy	Speed Whizz Uppers Billy	Speed Whizz Uppers Billy	Speed Whizz Uppers Billy	Speed Whizz Uppers Billy
Cannabis	Marijuana Grass Hash Ganja Blow Draw Skunk	Marijuana Grass Hash Ganja Blow Draw Skunk	Marijuana Grass Hash Ganja Blow Draw Skunk Weed Spliff	Marijuana Grass Hash Ganja Blow Draw Skunk Weed Spliff	Marijuana Grass Hash Ganja Blow Draw Skunk Weed Spliff	Marijuana Grass Hash Ganja Blow Draw Skunk Weed Spliff	Marijuana Grass Hash Ganja Blow Draw Skunk Weed Spliff	Marijuana Grass Hash Ganja Blow Draw Skunk Weed Spliff	Marijuana Grass Hash Ganja Blow Draw Skunk Weed Spliff
Cocaine	Coke	Coke	Coke	Coke	Coke	Coke	Coke	Coke	Coke
Crack	Rock, Stones	Rock, Stones	Rock, Stones	Rock, Stones	Rock, Stones	Rock, Stones	Rock, Stones	Rock, Stones	Rock, Stones
Ecstasy	'E'	'E'	'E'	'E'	'E'	'E'	'E'	'E'	'E'
Heroin	Smack Skag 'H'	Smack Skag 'H'	Smack Skag 'H' Brown	Smack 'H' Brown	Smack 'H' Brown	Smack 'H' Brown	Smack 'H' Brown	Smack 'H' Brown	Smack 'H' Brown
LSD/Acid	LSD/Acid	LSD/Acid	LSD/Acid	LSD/Acid	LSD/Acid	LSD/Acid	LSD/Acid	LSD/Acid	LSD/Acid
Magic Mushrooms	Magic Mushrooms	Magic Mushrooms	Magic Mushrooms	Magic Mushrooms	Magic Mushrooms	Magic Mushrooms	Magic Mushrooms	Magic Mushrooms	Magic Mushrooms
Methadone/ Physeptone	Methadone/ Physeptone	Methadone/ Physeptone	Methadone/ Physeptone	Methadone/ Physeptone	Methadone/ Physeptone	Methadone/ Physeptone	Methadone/ Physeptone	Methadone/ Physeptone	Methadone/ Physeptone
Semeran	Semeran	Semeran	Semeran	Semeran	Semeran	Semeran	Semeran	Semeran	Semeran
Tranquillizers	Tranquillizers Temazepam Valium	Tranquillizers Temazepam Valium	Tranquillizers Temazepam Valium	Tranquillizers Temazepam Valium	Tranquillizers Temazepam Valium	Tranquillizers Temazepam Valium	Tranquillizers Temazepam Valium	Tranquillizers Temazepam Valium	Tranquillizers Temazepam Valium
Amyl Nitrite	Poppers	Poppers	Poppers	Poppers	Poppers	Poppers	Poppers	Poppers	Poppers
Anabolic steroids	Steroids	Steroids	Steroids	Steroids	Steroids	Steroids	Steroids	Steroids	Steroids
Ketamine								Ketamine	Ketamine
Glues	Glues, Solvents, Gas, Aerosols	Glues, Solvents, Gas, Aerosols	Glues, Solvents, Gas, Aerosols	Glues, Solvents, Gas, Aerosols	Glues, Solvents, Gas, Aerosols	Glues, Solvents, Gas, Aerosols	Glues, Solvents, Gas, Aerosols	Glues, Solvents, Gas, Aerosols	Glues, Solvents, Gas, Aerosols

CHAPTER 6 - Methodological limitations

It must be noted that the figures derived from the BCS are *estimates*. As with any sample survey, the BCS estimates are subject to sampling error and a range of other methodological limitations.

SAMPLING ERROR

A sample, as used in the BCS, is a small-scale representation of the population from which it is drawn. As such, the sample may produce estimates that differ from figures that would have been obtained if the whole population had been interviewed. The size of the error depends on the size and design of the survey and the size and variability of the estimate. It can be computed and used to construct confidence intervals. Sampling error is also taken into account in tests of statistical significance.

Although, the BCS is large by the standards of most surveys, its estimates will be imprecise, particularly for rare crimes such as robbery and serious assault. Therefore, differences between figures (e.g. between the risk of violence for one group in the population and the risk for another) should generally only be reported if they are statistically significant, i.e. take sampling error into account.

NON-RESPONSE

As in any voluntary survey, the BCS is subject to non-response error. The BCS has managed to maintain a response rate (75%) that is high compared with other similar household surveys. However, non-response has implications for the measurement of crime if non-respondents have different experiences of victimisation to respondents. For example, young men tend to be less likely to respond but have higher than average risks of victimisation. Conversely, non-victims may be less likely to respond because they have less interest in the survey than victims. To explore the implications of non-response on survey estimates those who refused to participate in the 1996 BCS were asked a short set of questions about their experiences of crime. The balance of the evidence is that, if anything, non-respondents face slightly lower crime risks (Lynn, 1997⁴).

The BCS adopted *calibration* weighting in 2001/02 to account for differing rates of non-response between people of different sex, ages and regions. Re-weighting using calibration weighting was carried out on all survey years back to and including 1996.

RECALL

The BCS asks respondents to recall their experiences of crime in the previous 12 months. The BCS crime measure is thus dependent on respondents' ability to accurately remember their experiences in the reference period. There are several issues which could prevent accurate recall:

- respondents could simply forget a relevant incident;
- respondents could remember an incident, but think it happened before the reference period and therefore not report it in the interview; conversely, respondents may remember an earlier incident as happening within the reference period and thus erroneously report it;

UNWILLINGNESS TO REPORT

Respondents may be unwilling to disclose victimisation experiences in a face-to-face interview setting. This is more likely to be the case for some crimes such as domestic violence (particularly

⁴ Lynn, P. (1997). 'Sampling Frame Effects on the British Crime Survey.' *Journal of the Royal Statistical Society*, 160, Part 2, 253-269.

if the offender is in the room during interview), rape and sexual assault⁵. Self-completion modules are used to collect sensitive information which alleviates this problem to some extent. Prevalence rates for domestic violence are around five times higher for self-completion modules than those obtained from the victim form (Nicholas et al., 2007⁶).

DEFINITIONS OF CRIME

Incidents which are legally offences may not be reported to the survey if the respondent does not view them as such. In an attempt to overcome this problem the BCS screener questions ask whether the respondent has experienced certain types of events. They do not refer to 'crimes', use legal terminology or refer to specific offences. The definitional problem is particularly relevant to minor incidents and some forms of violence. Moreover, different social groups may have different perceptions of what does and does not constitute an incident. Evidence suggests that better-off groups have a lower threshold of tolerance and are therefore more likely to report minor incidents to the survey (Sparks et al., 1977⁷).

Overall methodological studies suggest that victimisation surveys are likely to undercount more trivial offences (e.g., vandalism and some assaults) and sensitive offences (e.g. sexual offences and domestic violence). Generally though, more serious incidents are likely to be over counted, as more salient events tend to be pulled forward in time.

⁵ Procedures are used to try to overcome this problem. The question is on a show card and interviewers are instructed that this section of the interview can be postponed if others are present during the interview.

⁶ Nicholas, S., Kershaw, C. and Walker, A. (2007) *Crime in England and Wales 2006/07*. Home Office Statistical Bulletin 11/07. London: Home Office. <http://www.homeoffice.gov.uk/rds/pdfs07/hosb1107.pdf>

⁷ Sparks, R. F., Genn, H. and Dodd, D. J. (1977). *Surveying victims*. London John Wiley.

CHAPTER 7 – BCS datasets

ACCESS TO BCS DATA

The BCS data is deposited at the UK Data Archive (UKDA). The UKDA hold the data for all years of the BCS. For each year there are at least two files. One holds the victimisation module data and the other the remaining parts of the questionnaire, with the exception of the self-completion modules (these are held separately and must be specifically requested for release⁸). See table below for details of all datasets deposited at the UKDA from 2006/07.

The victimisation module data is kept as a separate file because it is a file based on incidents not people. All other parts of the questionnaire are based on respondents - that is they have a case by variable structure in which each case is a unique individual. In contrast the Victim File is based on incidents reported by respondents, thus only those who have been a victim will appear in this file.

External researchers who would like to conduct secondary analysis of BCS data should contact the Economic and Social Data Service at www.esds.ac.uk or, UK Data Archive, University of Essex, Colchester, CO4 3SQ. Users of BCS data are requested to supply the UKDA with the bibliographic details of any published work.

CHANGES FROM 2006-07

For the first time the main BCS 2006/07 dataset includes aggregated offence data from the Victim File (see Appendix C for classification of incidents), alongside the usual demographic information and all the data collected from the core questionnaire and associated modules. This means users will be able to create incidence and prevalence rates, and look at repeat victimisation, using variables which already exist on the dataset.

From 2006/07, the self-completion drugs data file will include those aged 16-24 from the young adult boost and those aged 16-59 from the main sample who have completed the module. The weight 'indivwgt' should be used when analysing 16-59 year olds and the weight variable 'ypcwgt' should be used when analysing just the 16-24 year old group.

From 2006/07 the Home Office will also be depositing a privilege access dataset which will contain lower geographical area variables than currently deposited. These will be available through a special access licence. More information on this data will be available shortly.

Table 2 below shows the naming conventions on data files that will be adopted from 2006-07 onwards.

⁸ Note that not all self completions are currently deposited because of confidentiality and anonymity issues.

Table 2: BCS Datasets deposited at UK Data Archive from 2006/07

<u>Name of dataset</u>	<u>Content</u>
BCS Ap ryy Ma ryy to ESRC	Main BCS data with additional derived variables
BCS Ap ryy Ma ryy VF to ESRC	Victim file
BCS Ap ryy Ma ryy privilege access to ESRC	Main BCS data with additional derived variables and lower geographical area variables
BCS Ap ryy Ma ryy - young adult - to ESRC	Main young adult data file – young adult boost cases only
BCS Ap ryy Ma ryy - VF young adult - to ESRC	Young adult victim file – young adult boost cases only
BCS Ap ryy Ma ryy - drugs sc module	Self-completion module on illicit drug use – 16-59 year olds from main sample and 16-24 year olds from young adult boost
BCS Ap ryy Ma ryy - drinking behaviour sc module	Self-completion module on drinking behaviour
BCS Ap ryy Ma ryy - stolen goods sc module	Self-completion module on stolen goods
BCS Ap ryy Ma ryy - inter-personal violence sc module	Self-completion module on domestic violence, sexual victimisation and stalking
BCS Ap ryy Ma ryy - sexual assaults sc module	Self-completion module on sexual assaults

CHAPTER 8 – BCS publications

BCS PUBLICATIONS

- **Crime in England & Wales** - A full statistical bulletin, published in July of each year, with detailed findings, including regional data. This is a pre-announced National Statistics publication. The latest of these reports, which also provides detailed information on police recorded crime figures, was published in July 2007, and can be found at: <http://www.homeoffice.gov.uk/rds/crimeew0607.html>
Factsheets summarising crime statistics for each of the Government Office Regions in England and for Wales can also be found at the link above.
- **Quarterly Update** - Shorter statistical updates produced on a quarterly basis, focusing specifically on victimisation rates and trends. These are web-only documents, based on BCS and police recorded crime, which cover headline statistics only. Again, a National Statistics publication.
- An annual bulletin covering drug misuse as reported on the BCS. The most recent bulletin for the 2006/07 BCS can be found at: <http://www.homeoffice.gov.uk/rds/pdfs07/hosb1807.pdf>

The above references are intended only to illustrate the types of reports and findings that are produced from the BCS. For more details on all RDS publications associated with the BCS see <http://www.homeoffice.gov.uk/rds/bcs1.html>

FORTHCOMING PUBLICATIONS

We will be publishing a regular series of National Statistics supplementary bulletins to accompany the key findings in the main annual report. The three supplementary bulletins will cover: serious crime; 'other' crime; and attitudes, perceptions and risks, and will report on additional analysis not included in the main annual publication with exact content to be agreed each year.

The first two supplementary reports in this new series have already been published:

Homicides, Firearms and Intimate Violence 2005/06
<http://www.homeoffice.gov.uk/rds/pdfs07/hosb0207.pdf>

Mobile phone theft, plastic card and identity fraud: Findings from the 2005/06 British Crime Survey
<http://www.homeoffice.gov.uk/rds/pdfs07/hosb1007.pdf>

We will also be publishing a regular set of tables throughout the reporting year which will be issued alongside the supplementary bulletins. Current plans for table updates are:

- Nature of crime tables
- Policing and confidence in the CJS
- Vandalism

The release of these tables will be pre-announced as National Statistics and will be available online at: <http://www.homeoffice.gov.uk/rds/bcs1.html>

APPENDIX A

Comparing the cycles: 1982 to 2006/07 BCS

	1982	1984	1988	1992	1994	1996	1998	2000	2001	2001/02	2002/03	2003/04	2004/05	2005/06	2006/07
Survey company	SCPR	NOP	SCPR/ NOP	SCPR	OPCS	SCPR	SCPR	SCPR & ONS	BMRB	BMRB	BMRB	BMRB	BMRB	BMRB	BMRB
Core sample size	10,905	11,030	10,392	10,059	14,520	16,348	14,947	19,411	8,973	32,787	36,450	37,931	45,120	47,796	47,203
Response rate	81%	77%	77%	77%	77%	83%	79%	74%	73%	73%	74%	75%	75%	75%	75%
Sampling frame ¹	ER	ER	ER	PAF	PAF	PAF	PAF	PAF	PAF	PAF	PAF	PAF	PAF	PAF	PAF
Ethnic boost sample	✗	✗	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓
Young adults boost sample	✗	✗	✗	✓	✗	✗	✗	✗	✓	✓	✓	✓	✓	✓	✓
CAPI/PAPI	PAPI	PAPI	PAPI	PAPI	CAPI	CAPI	CAPI	CAPI	CAPI	CAPI	CAPI	CAPI	CAPI	CAPI	CAPI
No. of victim forms	4	4	4	5	5	6	6	6	6	6	6	6	6	6	6
Self-completion element	✗	✗	✗	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
Type of sample	Over sampled in inner city areas							Proportional sample	Over sampled in less populous PFAs (minimum=600)				Over sampled in less populous PFAs (minimum=1000)		

ER - Electoral Register

PAF - Small Users Postcode Address File

PAPI - Paper and Pencil Interviewing

CAPI - Computer Assisted Personal Interviewing

OPCS merged with the Central Statistical Office (CSO) in 1998 to form the Office for National Statistics (ONS).

SCPR - Social and Community Planning Research has now changed its name to the National Centre for Social Research.

BMRB - British Market Research Bureau

APPENDIX B

Design Factors applied to crime measures in 2006/07

<u>Crime measures</u>	<u>Design Factor</u>
<u>Incidents</u>	
All household offences	1.38
Burglary	1.16
All vehicle thefts (all households)	1.22
All household acquisitive	1.29
Vandalism	1.29
Vehicle vandalism	1.26
Other vandalism	1.17
Bicycle theft	1.22
Other household theft	1.22
All personal (not including sex)	1.37
All BCS violence (no snatch theft)	1.35
with injury (no snatch theft)	1.32
with no injury (no snatch theft)	1.39
All personal acquisitive	1.44
Thefts from the person	1.39
Robbery	1.63
Other thefts of personal property	1.37
<u>Prevalence</u>	
Burglary	1.22
All vehicle thefts (owners)	1.29
All vehicle thefts (all households)	1.22
Vandalism	1.29
All household crime	1.38
Theft from the person	1.39
Violent crime (no snatch theft)	1.37
<i>with injury (no snatch theft)</i>	1.34
<i>with no injury (no snatch theft)</i>	1.44
All personal crime	1.42
All BCS crime	1.53
<u>Fear of crime</u>	
Burglary	1.50
Car crime	1.42
Violent crime	1.70
<u>Disorder</u>	
Perceived level of ASB	1.78
Disorder(1) teenagers hanging around	1.71
Disorder(2) vandalism, graffiti etc	1.70
Disorder(4) people using/dealing drugs	1.88
Disorder(5) people being drunk or rowdy	1.79
Disorder(6) noisy neighbours	1.50
Disorder(7) litter/rubbish	1.71

Disorder(8) abandoned cars	1.75
----------------------------	------

Confidence in criminal justice system

Respects the rights of people accused of committing a crime and treats them fairly	1.41
--	------

Effective in bringing people who commit crimes to justice	1.39
---	------

Deals with cases promptly and efficiently	1.40
---	------

Reducing crime	1.43
----------------	------

Meets the needs of victims of crime	1.40
-------------------------------------	------

Dealing with young people accused of crime	1.33
--	------

Treats witnesses well	1.37
-----------------------	------

Confidence in local police

Police in local area doing good or excellent job	1.41
--	------

APPENDIX C

Classification of incidents – crime codes

vandalism (vandalis)

- 80 Arson
- 81 Criminal damage to a motor vehicle (£20 or under)
- 82 Criminal damage to a motor vehicle (over £20)
- 83 Criminal damage to the home (£20 or under)
- 84 Criminal damage to the home (over £20)
- 85 other criminal damage (£20 or under)
- 86 other criminal damage (over £20)

vehicle vandalism (mv.vand)

- 81 Criminal damage to a motor vehicle (£20 or under)
- 82 Criminal damage to a motor vehicle (over £20)

other vandalism (homevand)

- 80 Arson
- 83 Criminal damage to the home (£20 or under)
- 84 Criminal damage to the home (over £20)
- 85 other criminal damage (£20 or under)
- 86 other criminal damage (over £20)

burglary (burglar)

- 51 burglary in a dwelling (nothing taken)
- 52 burglary in a dwelling (something taken)
- 53 attempted burglary in a dwelling

burglary - attempted (burgatts)

- 53 attempted burglary in a dwelling

burglary - attempts and no loss (burgatno)

- 51 burglary in a dwelling (nothing taken)
- 53 attempted burglary in a dwelling

burglary - with entry (burgentr)

- 51 burglary in a dwelling (nothing taken)
- 52 burglary in a dwelling (something taken)

burglary - with loss (burgloss)

- 52 burglary in a dwelling (something taken)

theft in a dwelling (theftdwe)

- 55 theft in a dwelling

theft from vehicle (theftfmv)

- 61 Theft from car/van
- 63 theft from motorbike, motorscooter or moped)

theft of vehicles (theftomv)

- 60 Theft of car/van
- 62 Theft of motorbike, motorscooter or moped

vehicle crime - attempts of and from (attmvthf)

- 71 attempted theft of/from car/van
- 72 attempted theft of/from motorcycle

all vehicle thefts (allmvthf)

- 60 Theft of car/van
- 61 Theft from car/van
- 62 Theft of motorbike, motorscooter or moped

63 theft from motorbike, motorscooter or moped
71 attempted theft of/from car/van
72 attempted theft of/from motorcycle

all vehicle crime (allmvcric)

60 Theft of car/van
61 Theft from car/van
62 Theft of motorbike, motorscooter or moped
63 theft from motorbike, motorscooter or moped
71 attempted theft of/from car/van
72 attempted theft of/from motorcycle
81 Criminal damage to a motor vehicle (£20 or under)
82 Criminal damage to a motor vehicle (over £20)

bicycle theft (bikethef)

64 theft of pedal cycle

other household theft (othhhctd)

50 attempted burglary to non-connected domestic garage/outhouse
55 theft in a dwelling
56 theft from a meter
57 burglary from non-connected domestic garage/outhouse-nothing taken
58 burglary from non-connected domestic garage/outhouse-something taken
65 Theft from outside dwelling (excl. theft of milk bottles)

comparable household crime (tohhcltd)

51 burglary in a dwelling (nothing taken)
52 burglary in a dwelling (something taken)
53 attempted burglary in a dwelling
60 Theft of car/van,
61 Theft from car/van
62 Theft of motorbike, motorscooter or moped
63 theft from motorbike, motorscooter or moped
64 theft of pedal cycle
71 attempted theft of/from car/van
72 attempted theft of/from motorcycle
80 Arson
81 Criminal damage to a motor vehicle (£20 or under)
82 Criminal damage to a motor vehicle (over £20)
83 Criminal damage to the home (£20 or under)
84 Criminal damage to the home (over £20)
85 other criminal damage (£20 or under)
86 other criminal damage (over £20)

all household offences (totalhh)

50 attempted burglary to non-connected domestic garage/outhouse
51 burglary in a dwelling (nothing taken)
52 burglary in a dwelling (something taken)
53 attempted burglary in a dwelling
55 theft in a dwelling
56 theft from a meter
57 burglary from non-connected domestic garage/outhouse-nothing taken
58 burglary from non-connected domestic garage/outhouse-something taken
60 Theft of car/van
61 Theft from car/van
62 Theft of motorbike, motorscooter or moped
63 theft from motorbike, motorscooter or moped
64 theft of pedal cycle
65 Theft from outside dwelling (excl. theft of milk bottles)
71 attempted theft of/from car/van,
72 attempted theft of/from motorcycle
80 Arson

- 81 Criminal damage to a motor vehicle (£20 or under)
- 82 Criminal damage to a motor vehicle (over £20)
- 83 Criminal damage to the home (£20 or under)
- 84 Criminal damage to the home (over £20)
- 85 other criminal damage (£20 or under)
- 86 other criminal damage (over £20)

acquisitive crimes (acquisit)

- 43 snatch theft from the person
- 44 other theft from the person
- 45 attempted theft from the person
- 51 burglary in a dwelling (nothing taken)
- 52 burglary in a dwelling (something taken)
- 53 attempted burglary in a dwelling
- 60 Theft of car/van
- 61 Theft from car/van
- 62 Theft of motorbike, motorscooter or moped
- 63 theft from motorbike, motorscooter or moped
- 64 theft of pedal cycle
- 71 attempted theft of/from car/van
- 72 attempted theft of/from motorcycle

sexual offences (sexoffen)

- 31 rape
- 34 attempted rape
- 35 indecent assault

common assault (commonas)

- 13 common assault
- 21 attempted assault

wounding (wounding)

- 11 serious wounding
- 12 other wounding
- 32 serious wounding with sexual motive
- 33 other wounding with sexual motive

robbery (robbery)

- 41 robbery
- 42 attempted robbery

theft from the person (theftper)

- 43 snatch theft from the person
- 44 other theft from the person
- 45 attempted theft from the person

theft from person & robbery (thfp.rob)

- 41 robbery
- 42 attempted robbery
- 43 snatch theft from the person
- 44 other theft from the person
- 45 attempted theft from the person

comparable violence (compvio)

- 13 common assault
- 21 attempted assault
- 11 serious wounding
- 12 other wounding
- 32 serious wounding with sexual motive
- 33 other wounding with sexual motive
- 41 robbery
- 42 attempted robbery

other thefts of personal property (othptheft)

67 other theft
73 other attempted theft

comparable personal (topthcls)

11 serious wounding
12 other wounding
32 serious wounding with sexual motive
33 other wounding with sexual motive
41 robbery
42 attempted robbery
43 snatch theft from the person
44 other theft from the person
45 attempted theft from the person

all personal including sex offences (totalper)

11 serious wounding
12 other wounding
13 common assault
21 attempted assault
31 rape
32 serious wounding with sexual motive
33 other wounding with sexual motive
34 attempted rape
35 indecent assault
41 robbery
42 attempted robbery
43 snatch theft from the person
44 other theft from the person
45 attempted theft from the person
67 other theft
73 other attempted theft

all personal not including sex offences (totperls)

11 serious wounding
12 other wounding
13 common assault
21 attempted assault
32 serious wounding with sexual motive
33 other wounding with sexual motive
41 robbery
42 attempted robbery
43 snatch theft from the person
44 other theft from the person
45 attempted theft from the person
67 other theft
73 other attempted theft

all assault (allassau)

11 serious wounding
12 other wounding
13 common assault
21 attempted assault
32 serious wounding with sexual motive
33 other wounding with sexual motive

robbery and wounding (violence)

11 serious wounding
12 other wounding
32 serious wounding with sexual motive
33 other wounding with sexual motive

41 robbery
42 attempted robbery

threats (threats)

91 threat to kill/assault made against, but not necessarily to respondent
92 sexual threat made against, but not necessarily to respondent
93 other threat or intimidation made against, but not necessarily to respondent
94 threats against others, made to the respondent

total BCS crime without sex offences (totalbcs)

11 serious wounding
12 other wounding
13 common assault
21 attempted assault
32 serious wounding with sexual motive
33 other wounding with sexual motive
41 robbery
42 attempted robbery
43 snatch theft from the person
44 other theft from the person
45 attempted theft from the person
50 attempted burglary to non-connected domestic garage/outhouse
51 burglary in a dwelling (nothing taken)
52 burglary in a dwelling (something taken)
53 attempted burglary in a dwelling
55 theft in a dwelling
56 theft from a meter
57 burglary from non-connected domestic garage/outhouse-nothing taken
58 burglary from non-connected domestic garage/outhouse-something taken
60 Theft of car/van
61 Theft from car/van
62 Theft of motorbike, motorscooter or moped
63 theft from motorbike, motorscooter or moped
64 theft of pedal cycle
65 Theft from outside dwelling (excl. theft of milk bottles)
67 other theft
71 attempted theft of/from car/van
72 attempted theft of/from motorcycle
73 other attempted theft
80 Arson
81 Criminal damage to a motor vehicle (£20 or under)
82 Criminal damage to a motor vehicle (over £20)
83 Criminal damage to the home (£20 or under)
84 Criminal damage to the home (over £20)
85 other criminal damage (£20 or under)
86 other criminal damage (over £20)

all BCS violence (allviol)

11 serious wounding
12 other wounding
13 common assault
21 attempted assault
32 serious wounding with sexual motive
33 other wounding with sexual motive
41 robbery
42 attempted robbery
43 snatch theft from the person

mugging (mugging1)

41 robbery
42 attempted robbery
43 snatch theft from the person

stealth theft from person (stealth)

- 44 other theft from the person
- 45 attempted theft from the person.

snatch theft from person (snatch)

- 43 snatch theft from the person

acquisitive crime against the household (hhacq)

- 50 attempted burglary to non-connected domestic garage/outhouse
- 51 burglary in a dwelling (nothing taken)
- 52 burglary in a dwelling (something taken)
- 53 attempted burglary in a dwelling
- 55 theft in a dwelling
- 56 theft from a meter
- 57 burglary from non-connected domestic garage/outhouse-nothing taken
- 58 burglary from non-connected domestic garage/outhouse-something taken
- 60 Theft of car/van
- 61 Theft from car/van
- 62 Theft of motorbike, motorscooter or moped
- 63 theft from motorbike, motorscooter or moped
- 64 theft of pedal cycle
- 65 Theft from outside dwelling (excl. theft of milk bottles)
- 71 attempted theft of/from car/van,
- 72 attempted theft of/from motorcycle

acquisitive crime against the individual (persacq)

- 41 robbery
- 42 attempted robbery
- 43 snatch theft from the person
- 44 other theft from the person
- 45 attempted theft from the person
- 67 other theft
- 73 other attempted theft

all BCS violence excluding robbery and snatch theft (violnors)

- 11 serious wounding
- 12 other wounding
- 13 common assault
- 21 attempted assault
- 32 serious wounding with sexual motive
- 33 other wounding with sexual motive

all BCS violence excluding snatch theft (violnos)

- 11 serious wounding
- 12 other wounding
- 13 common assault
- 21 attempted assault
- 32 serious wounding with sexual motive
- 33 other wounding with sexual motive
- 41 robbery
- 42 attempted robbery

APPENDIX D

Home Office derived variables for the main 2006/07 BCS datafile

*AGE OF RESPONDENT.

recode age (16 thru 29=1)(30 thru 59=2)(60 thru 99=3)(else=sysmis) into agegrp.

variable labels agegrp 'Age group (3 bands)'.

value labels agegrp

- 1 '16-29'
- 2 '30-59'
- 3 '60+'.

recode age (16 thru 19=1)(20 thru 24=2)(25 thru 34=3)(35 thru 44 =4)(45 thru 54=5)
(55 thru 64=6)(65 thru 74=7)(75 thru 84=8)(85 thru 99=9)(else=sysmis) into agelong.

variable labels agelong 'Age group (9 bands)'.

value labels agelong

- 1 '16-19'
- 2 '20-24'
- 3 '25-34'
- 4 '35-44'
- 5 '45-54'
- 6 '55-64'
- 7 '65-74'
- 8 '75-84'
- 9 '85+'.

recode age (16 thru 24=1)(25 thru 44 =2)(45 thru 64=3)(65 thru 74=4)(75 thru 99=5)(else=sysmis) into
ageshort.

variable labels ageshort 'Age group (5 bands)'.

value labels ageshort

- 1 '16-24'
- 2 '25-44'
- 3 '45-64'
- 4 '65-74'
- 5 '75+'.

recode age (16 thru 24=1) (25 thru 34 =2) (35 thru 44 = 3) (45 thru 54 = 4) (55 thru 64 = 5) (65 thru 74 = 6)
(75 thru 99 = 7) (else=sysmis) into agegrp7.

variable labels agegrp7 'Age group (7 bands)'.

value labels agegrp7

- 1 '16-24'
- 2 '25-34'
- 3 '35-44'
- 4 '45-54'
- 5 '55-64'
- 6 '65-74'
- 7 '75+'.

* AGE VARIABLES FOR CALIBRATION.

recode age (0 thru 4=1)(5 thru 9=2)(10 thru 15=3)(16 thru 19=4)(20 thru 24=5)(25 thru 29=6)(30 thru 34
=7)(35 thru 39=8)

(40 thru 44=9)(45 thru 49 = 10)(50 thru 54=11)(55 thru 59=12)(60 thru 64 =13)(65 thru 69=14)

(70 thru 74=15)(75 thru 79=16)(80 thru 99=17)(else=sysmis) into AgeCalib.

variable labels AgeCalib 'Age group (17 bands, for calibration)'.

value labels AgeCalib

- 1 '0-4'
- 2 '5-9'
- 3 '10-15'
- 4 '16-19'
- 5 '20-24'
- 6 '25-29'

7 '30-34'
8 '35-39'
9 '40-44'
10 '45-49'
11 '50-54'
12 '55-59'
13 '60-64'
14 '65-69'
15 '70-74'
16 '75-79'
17 '80+'.

*RESPONDENT OF WORKING AGE (TO RESTRICT ANALYSIS TO THOSE OF WORKING AGE).

recode age (16 thru 59=1) into wrkage.
if ((sex=2) and (wrkage=1)) rworkage=1.
recode age (16 thru 64=1) into wrkage.
if ((sex=1) and (wrkage=1)) rworkage=1.
variable labels rworkage 'Respondent of working age'.
value labels rworkage
1 'Respondent of working age'.

*AGE/SEX OF RESPONDENT.

Compute sexage = (sex-1)*3 + agegrp.

variable labels sexage 'Age within Sex (3 bands each)'.
value labels sexage
1 'Male 16-29'
2 'Male 30-59'
3 'Male 60+.'
4 'Female 16-29'
5 'Female 30-59'
6 'Female 60+'.

Compute sexage1 = (sex-1)*5 + ageshort.

variable labels sexage1 'Age within Sex (6 bands each)'.
value labels sexage1
1 'Male 16-24'
2 'Male 25-44'
3 'Male 45-64'
4 'Male 65-74'
5 'Male 75+.'
6 'Female 16-24'
7 'Female 25-44'
8 'Female 45-64'
9 'Female 65-74'
10 'Female 75+'.

Compute sexage14 = (sex-1)*7 + agegrp7.

variable labels sexage14 'Age within Sex (7 bands each)'.
value labels sexage14
1 'Male 16-24'
2 'Male 25-34'
3 'Male 35-44'
4 'Male 45-54'
5 'Male 55-64'
6 'Male 65-74'
7 'Male 75+.'
8 'Female 16-24'
9 'Female 25-34'

10 'Female 35-44'
11 'Female 45-54'
12 'Female 55-64'
13 'Female 65-74'
14 'Female 75+'.

*RESPONDENT ETHNICITY - USES ONS LEVEL 2 CLASSIFICATION.

*WHITE, MIXED, ASIAN, BLACK AND CHINESE AND OTHER.

recode ethnic (1 thru 3=1) (4 thru 7=2) (8 thru 11=3) (12 thru 14=4) (15 thru 16=5) (else=sysmis) into ethgrp2.

variable labels ethgrp2 'Ethnic Group (5 categories)'.

value labels ethgrp2

1 'White'
2 'Mixed'
3 'Asian or Asian British'
4 'Black or Black British'
5 'Chinese or Other'.

recode ethgrp2 (1=1) (2 thru 5=2) (else=sysmis) into ethgrp3.

variable labels ethgrp3 'Ethnic group (white/non-white)'.

value labels ethgrp3

1 'White'
2 'Non-white'.

recode ethgrp2 (1=1) (3=2) (4=3) (5=4) (2=4) (else=sysmis) into ethgrp4.

variable labels ethgrp4 'Ethnic Group (4 categories)'.

value labels ethgrp4

1 'White'
2 'Asian or Asian British'
3 'Black or Black British'
4 'Other (incl Chinese and Mixed)'.

recode ethnic (1 thru 3=1) (4 thru 7=2) (8 thru 11=3) (12 thru 14=4) (15 thru 16=5) (98,99=6) (else=sysmis) into ethgrp5.

variable labels ethgrp5 'Ethnic Group (5 categories - harmonised)'.

value labels ethgrp5

1 'White'
2 'Mixed'
3 'Asian or Asian British'
4 'Black or Black British'
5 'Chinese or Other'
6 'Not stated'.

*HOUSEHOLD REFERENCE PERSON ETHNICITY - USES ONS LEVEL 2 CLASSIFICATION.

*For household offences it is the ethnicity of the HRP that is needed, not that of the respondent.

compute ethnhead=0.

*HRP is respondent.

recode ethnic (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) into ethnic1a.

if (whohrp=1) ethnhead=ethnic1a.

recode ethnic (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) (98,99=6) into ethnic1b.

if (whohrp=1) ethnhead2=ethnic1b.

*HRP is person 2.

recode ethnic2 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) into ethnic2a.

if (whohrp2=1) ethnhead=ethnic2a.

recode ethnic2 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) (98,99=6) into ethnic2b.

if (whohrp2=1) ethnhead2=ethnic2b.

*HRP is person 3.

recode ethnic3 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) into ethnic3a.

if (whohrp3=1) ethnhead=ethnic3a.
recode ethnic3 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) (98,99=6) into ethnic3b.
if (whohrp3=1) ethnhead2=ethnic3b.

*HRP is person 4.
recode ethnic4 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) into ethnic4a.
if (whohrp4=1) ethnhead=ethnic4a.
recode ethnic4 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) (98,99=6) into ethnic4b.
if (whohrp4=1) ethnhead2=ethnic4b.

*HRP is person 5.
recode ethnic5 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) into ethnic5a.
if (whohrp5=1) ethnhead=ethnic5a.
recode ethnic5 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) (98,99=6) into ethnic5b.
if (whohrp5=1) ethnhead2=ethnic5b.

*HRP is person 6.
recode ethnic6 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) into ethnic6a.
if (whohrp6=1) ethnhead=ethnic6a.
recode ethnic6 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) (98,99=6) into ethnic6b.
if (whohrp6=1) ethnhead2=ethnic6b.

*HRP is person 7.
recode ethnic7 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) into ethnic7a.
if (whohrp7=1) ethnhead=ethnic7a.
recode ethnic7 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) (98,99=6) into ethnic7b.
if (whohrp7=1) ethnhead2=ethnic7b.

*HRP is person 8.
recode ethnic8 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) into ethnic8a.
if (whohrp8=1) ethnhead=ethnic8a.
recode ethnic8 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) (98,99=6) into ethnic8b.
if (whohrp8=1) ethnhead2=ethnic8b.

*HRP is person 9.
recode ethnic9 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) into ethnic9a.
if (whohrp9=1) ethnhead=ethnic9a.
recode ethnic9 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) (98,99=6) into ethnic9b.
if (whohrp9=1) ethnhead2=ethnic9b.

*HRP is person 10.
recode ethnic10 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) into ethnc10a.
if (whohrp10=1) ethnhead=ethnc10a.
recode ethnic10 (1,2,3=1) (4, 5, 6, 7=2) (8, 9, 10, 11=3) (12, 13, 14 =4) (15, 16=5) (98,99=6) into ethnc10b.
if (whohrp10=1) ethnhead2=ethnc10b.

variable labels ethnhead 'Ethnic Group of HRP (5 categories)'.
value labels ethnhead

- 1 'White'
- 2 'Mixed'
- 3 'Asian or Asian British'
- 4 'Black or Black British'
- 5 'Chinese or other'.

variable labels ethnhead2 'Ethnic Group of HRP (5 categories - harmonised)'.
value labels ethnhead2

- 1 'White'
- 2 'Mixed'
- 3 'Asian or Asian British'
- 4 'Black or Black British'
- 5 'Chinese or other'
- 6 'Not stated'.

*HOUSEHOLD ETHNICITY.

```
recode ethnic ethnic2 ethnic3 ethnic4 ethnic5 ethnic6 ethnic7 ethnic8 ethnic9 ethnic10 (1 thru 3=1) (4 thru 16=0)
  into eth1 eth2 eth3 eth4 eth5 eth6 eth7 eth8 eth9 eth10.
compute anyeth=0.
if ((eth1=0) or (eth2=0) or (eth3=0) or (eth4=0) or (eth5=0) or (eth6=0) or (eth7=0) or (eth8=0) or (eth9=0) or (eth10=0)) anyeth=1.
variable labels anyeth 'Any member of household BME'.
value labels anyeth
  0 'No BME household members'
  1 'One or more household members BME'.
```

*RESPONDENT RELIGION.

*Jewish group in 'other' category.

```
Recode relig2 (1=1) (2=2) (3=3) (4=6) (5=4) (6=5) (7=6) (8=7) (else=sysmis) into relig2a.
Variable labels relig2a 'Respondent religion (7 categories)'.
value labels relig2a
  1 'Christian'
  2 'Buddhist'
  3 'Hindu'
  4 'Muslim'
  5 'Sikh'
  6 'Other'
  7 'No religion'.
```

*Jewish and sikh groups in 'other' category.

```
Recode relig2 (1=1) (2=2) (3=3) (4=5) (5=4) (6=5) (7=5) (8=6) (else=sysmis) into relig3.
Variable labels relig3 'Respondent religion (6 categories)'.
value labels relig3
  1 'Christian'
  2 'Buddhist'
  3 'Hindu'
  4 'Muslim'
  5 'Other'
  6 'No religion'.
```

*HOUSEHOLD REFERENCE PERSON RELIGION.

```
compute hrprelnw =relig2.
```

```
if (hrp=2 and hrprelg2=1) hrprelnw=1.
if (hrp=2 and hrprelg2=2) hrprelnw=2.
if (hrp=2 and hrprelg2=3) hrprelnw=3.
if (hrp=2 and hrprelg2=4) hrprelnw=4.
if (hrp=2 and hrprelg2=5) hrprelnw=5.
if (hrp=2 and hrprelg2=6) hrprelnw=6.
if (hrp=2 and hrprelg2=7) hrprelnw=7.
if (hrp=2 and hrprelg2=8) hrprelnw=8.
if (hrp=2 and hrprelg2=98) hrprelnw=98.
if (hrp=2 and hrprelg2=99) hrprelnw=99.
variable labels hrprelnw 'HRP religion - combined categories'.
value labels hrprelnw
  1 'Christian'
  2 'Buddhist'
  3 'Hindu'
  4 'Jewish'
  5 'Muslim'
  6 'Sikh'
  7 'Other'
```

8 'No religion'.

*Jewish group in 'other' category.

recode hrprelnw (1=1) (2=2) (3=3) (4=6) (5=4) (6=5) (7=6) (8=7) (else=sysmis) into hrprel2a.

variable labels hrprel2a 'HRP religion (7 categories)'.

value labels hrprel2a

- 1 'Christian'
- 2 'Buddhist'
- 3 'Hindu'
- 4 'Muslim'
- 5 'Sikh'
- 6 'Other'
- 7 'No religion'.

*Jewish and sikh groups in 'other' category.

recode hrprelnw (1=1) (2=2) (3=3) (4=5) (5=4) (6=5) (7=5) (8=6) (else=sysmis) into hrprelg3.

variable labels hrprelg3 'HRP religion (6 categories)'.

value labels hrprelg3

- 1 'Christian'
- 2 'Buddhist'
- 3 'Hindu'
- 4 'Muslim'
- 5 'Other'
- 6 'No religion'.

*RESPONDENT OCCUPATION CODING.

recode respsec2 (1.1=1) (1.2=1) (2=1) (3=2) (4=3) (5=4) (6 thru 7=5) (8=6) (else=sysmis) into rnssec5.

variable labels rnssec5 'Occupation coding of Respondent (5 categories)'.

value labels rnssec5

- 1 'Managerial and professional occs'
- 2 'Intermediate occs'
- 3 'Small employers and own account workers'
- 4 'Lower supervisory and technical occs'
- 5 'Semi-routine and routine occs'
- 6 'Never worked and long term unemployed'.

recode respsec2 (1.1=1) (1.2=1) (2=1) (3=2) (4=3) (5=4) (6 thru 7=5) (8=6) (9=7) (else=sysmis) into rnssec6.

variable labels rnssec6 'Occupation coding of Respondent (5 categories & not classified)'.

value labels rnssec6

- 1 'Managerial and professional occs'
- 2 'Intermediate occs'
- 3 'Small employers and own account workers'
- 4 'Lower supervisory and technical occs'
- 5 'Semi-routine and routine occs'
- 6 'Never worked and long term unemployed'
- 7 'Not classified'.

recode respsec (1.0,2.0,3.1,3.2,3.3,3.4,4.1,4.2,4.3,4.4,5.0,6.0=1) (7.1,7.2,7.3,7.4=2) (8.1,8.2,9.1,9.2=3) (10.0,11.1,11.2=4) (12.1,12.2,12.3,12.4,12.5,12.6,12.7,13.1,13.2,13.3,13.4,13.5=5) (14.1,14.2=6) (15.0=7) (16.0,17.0=8)

(else=sysmis) into rnssec6a.

variable labels rnssec6a 'Occupation coding of Respondent (5 categories & students & not classified)'.

value labels rnssec6a

- 1 'Managerial and professional occs'
- 2 'Intermediate occs'
- 3 'Small employers and own account workers'
- 4 'Lower supervisory and technical occs'
- 5 'Semi-routine and routine occs'
- 6 'Never worked and long term unemployed'
- 7 'Full-time students'

8 'Not classified'.

recode respsec2 (1.1=1) (1.2=1) (2=1) (3 thru 4=2) (5 thru 7=3) (8=4) (else=sysmis) into rnssec3.

variable labels rnssec3 'Occupation coding of Respondent (3 categories)'.

value labels rnssec3

1 'Managerial and professional occs'

2 'Intermediate occs'

3 'Routine and manual occs'

4 'Never worked and long term unemployed'.

recode respsec2 (1.1=1) (1.2=1) (2=1) (3 thru 4=2) (5 thru 7=3) (8=4) (9=5) (else=sysmis) into rnssec4.

variable labels rnssec4 'Occupation coding of Respondent (3 categories & not classified)'.

value labels rnssec4

1 'Managerial and professional occs'

2 'Intermediate occs'

3 'Routine and manual occs'

4 'Never worked and long term unemployed'

5 'Not classified'.

recode respsec (1.0,2.0,3.1,3.2,3.3,3.4,4.1,4.2,4.3,4.4,5.0,6.0=1) (7.1,7.2,7.3,7.4,8.1,8.2,9.1,9.2=2)
(10.0,11.1,11.2,12.1,12.2,12.3,12.4,12.5,12.6,12.7,13.1,13.2,13.3,13.4,13.5=3) (14.1,14.2=4) (15.0=5)
(16.0,17.0=6) (else=sysmis) into rnssec4a.

variable labels rnssec4a 'Occupation coding of Respondent (3 categories & students & not classified)'.

value labels rnssec4a

1 'Managerial and professional occs'

2 'Intermediate occs'

3 'Routine and manual occs'

4 'Never worked and long term unemployed'

5 'Full-time students'

6 'Not classified'.

*RESPONDENT MARITAL STATUS.

recode marst (1=3) (2=1) (3=6) (4=5) (5=4) (else=sysmis) into marital .

if (cohab=1 or cohab=3) marital=2.

variable labels marital 'Respondent Marital status'.

value labels marital

1 'Married'

2 'Cohabiting'

3 'Single'

4 'Widowed'

5 'Divorced'

6 'Separated'.

recode marital (1,2 =1) (3=2) (4=3) (5 thru 6=4) (else=sysmis) into margrp.

variable labels margrp 'Respondent de facto marital status'.

value labels margrp

1 'Married or de facto'

2 'Single'

3 'Widowed'

4 'Separated or divorced'.

*ONS HARMONISED MARITAL VARIABLES.

recode marst (1=3) (2=1) (3=4) (4=5) (5=6) (else=sysmis) into livharm1 .

if (cohab=1 or cohab=3) livharm1=2.

variable labels livharm1 'ONS harmonised marital status'.

value labels livharm1

1 'Married'

2 'Cohabiting'

3 'Single'

4 'Separated'

5 'Divorced'
6 'Widowed'.

recode livharm1 (1,2=1) (3,4,5,6=2) (else=sysmis) into livharm2 .

variable labels livharm2 'Whether respondent living in a couple'.
value labels livharm2
1 'Living in a couple'
2 'Not living in a couple'.

*ADULT/CHILD COUNTS.

recode nadults (1=1) (2 thru 10=0) (else=sysmis) into nad2.
variable labels nad2 'One adult household identifier'.
value labels nad2
0 'More than one adult in household'
1 'One adult household'.

recode nchil (0=0) (1 thru 10=1) (else=sysmis) into nchil2.
variable labels nchil2 'Children in household identifier'.
value labels nchil2
0 'No children in household'
1 'Children in household'.

*HEALTH AND ILLNESS - ONS HARMONISED VARIABLE.

if (illness=1) Lillharm=2.
if (illness=2) Lillharm=1.
variable labels Lillharm 'Disability/long-standing illness (2 categories)'.
value labels Lillharm
1 'No long standing illness'
2 'Long standing illness'.

if (illness=2) ill=1. /* JH */
if (illness=1 and limits=2) ill=2.
if (illness=1 and limits=1) ill=3.
variable labels ill 'Disability/long-standing illness (3 categories)'.
value labels ill
1 'No disability/illness'
2 'Non-limiting disability/illness'
3 'Limiting disability/illness'.

recode genhealt (1 thru 5=copy) (else=sysmis) into genheal2.
variable labels genheal2 'Health (all categories)'.
value labels genheal2
1 'Very good'
2 'Good'
3 'Fair'
4 'Poor'
5 'Very poor' .

recode genhealt (1,2 =1) (3=2) (4,5 =3) into health2.
variable labels health2 'Health (3 categories)'.
value labels health2
1 'Good (incl very good)'
2 'Fair'
3 'Poor (incl very poor)'.

*EDUCATION.

compute educat3=0.
if (educint=2) educat3=1.
if ((educat2=6) or (educat2=7)) educat3=2.

```

if ((educat2=4) or (educat2=5)) educat3=3.
if ((educat2=1) or (educat2=2) or (educat2=3)) educat3=4.
if (educat2=8) educat3=5.
recode educat3 (1 thru 5=copy) (else=sysmis).
variable labels educat3 'Respondent education (5 categories)'.
value labels educat3
  1 'None'
  2 'O level/GCSE'
  3 'Apprenticeship or A/AS level'
  4 'Degree or diploma'
  5 'Other' .

```

```

recode educat2 (1 thru 4 = 1) (5 thru 8 = 2) (else=sysmis) into educat4.
if (educat2=2) educat4=3.
variable labels educat4 'Respondent education (3 categories)'.
value labels educat4
  1 'A levels or above'
  2 'below A-level'
  3 'No qualifications'.

```

*NEWSPAPER READERSHIP.

```

recode newspap2 (5,6,7,8,10=1) (1,2,3,4,9=2) (11 thru 13=3) (14=4) (else=sysmis) into newspap2. variable
labels newspap2 'Newspaper readership'.
value labels newspap2
  1 'National broadsheets'
  2 'National tabloids'
  3 'Other daily newspaper'
  4 'More than one'.

```

*LENGTH OF TIME AT ADDRESS - ONS HARMONISED.

```

Compute ysadharm = yrsaddr .
variable labels ysadharm 'ONS harmonised length of time at address'.
value labels ysadharm
  1 'Less than 12 months'
  2 '12 months, less than 2 years'
  3 '2 years, less than 3 years'
  4 '3 years, less than 5 years'
  5 '5 years, less than 10 years'
  6 '10 years, less than 20 years'
  7 '20 years or more'.

```

```

recode yrsaddr (1=1) (2=2) (3,4=3) (5=4) (6,7=5) into yrsaddr2.
variable labels yrsaddr2 'Length of time at address'.
value labels yrsaddr2
  1 'Less than one year'
  2 '1-2 years'
  3 '2-5 years'
  4 '5-10 years'
  5 'Longer'.

```

*RESPONDENT/HRP INDICATOR.

```

if (nadults=1 or whohrp=1) hrp=1.
if (whohrp=2) hrp=2.
variable labels hrp 'Household Reference Person Indicator'.
value labels hrp
  1 'Respondent is HRP'
  2 'Respondent is not HRP'.

```

*HOUSEHOLD INCOME.


```

recode tothhin1 (1,13=1) (2=2) (3=3) (4=4) (5=5) (6,7,8,9,10,11,12=6) (else=sysmis) into hhinc6. variable
labels hhinc6 'Total household income (6 bands)'.
value labels hhinc6
  1 'Under £2,500'
  2 '£2,500-£4,999'
  3 '£5,000-£9,999'
  4 '£10,000-£14,999'
  5 '£15,000-£19,999'
  6 '£20,000 or more'.

recode hhinc6 (1,2=1) (3,4=2) (5=3) (6=4) (else=sysmis) into hhinc4.
variable labels hhinc4 'Total household income (4 bands)'.
value labels hhinc4
  1 'Under £5,000'
  2 '£5,000-£14,999'
  3 '£15,000-£19,999'
  4 '£20,000 or more'.

recode tothhin1 (1,2,3,13=1) (4=2) (5=3) (6,7=4) (8,9,10,11,12=5) (else=sysmis) into hhinc5.
variable labels hhinc5 'Total household income (5 bands)'.
value labels hhinc5
  1 'Under £10,000'
  2 '£10,000-£14,999'
  3 '£15,000-£19,999'
  4 '£20,000-£29,999'
  5 '£30,000 or more'.

compute tothhinc=0.
recode tothhin1 (1=1) (2=2) (3=3) (4=4) (5=5) (6,7=6) (8,9,10,11=7) (12=8) (13=9) (98=98) (99=99)
(else=sysmis) into tothhinc.
variable labels tothhinc 'Total household income (8 bands & spontaneous)'.
value labels tothhinc
  1 'Under £2,500'
  2 '£2,500-£4,999'
  3 '£5,000-£9,999'
  4 '£10,000-£14,999'
  5 '£15,000-£19,999'
  6 '£20,000-£29,999'
  7 '£30,000-£49,999'
  8 '£50,000 or more'
  9 'spontaneous'.

recode tothhin1 (1,2=1) (3=2) (4,5=3) (6,7=4) (8,9,10,11,12=5) (13=6) (else=0) into tothhin2.
variable labels tothhin2 'Total household income (5 bands & spontaneous)'.
value labels tothhin2
  1 'Under £5,000'
  2 '£5,000-£9,999'
  3 '£10,000-£19,999'
  4 '£20,000-£29,999'
  5 '£30,000 or more'
  6 'spontaneous'.

recode tothhin2 (6,1=1) (2 thru 5=copy) (else=sysmis) into tothhin3.
variable labels tothhin3 'Total household income (5 bands)'.
value labels tothhin3
  1 'Under £5,000 (incl spontaneous)'
  2 '£5,000-£9,999'
  3 '£10,000-£19,999'
  4 '£20,000-£29,999'
  5 '£30,000 or more'.

compute tothhin3=0.
recode tothhin1 (1 thru 3=1) (4,5=2) (6,7=3) (8,9=4) (10,11=5) (12=6) (13=7) (else=sysmis) into tothhin3.

```

variable labels tothhin3 'Total household income (6 bands & spontaneous)'.

value labels tothhin3

- 1 'Under £10,000'
- 2 '£10,000-£19,999'
- 3 '£20,000-£29,999'
- 4 '£30,000-£39,999'
- 5 '£40,000-£49,999'
- 6 '£50,000 or more'
- 7 'Nothing/No work/scheme'.

*TENURE - ONS HARMONISED VARIABLE.

if (tenure1=1 or tenure1=2 or tenure1=3) tenharm=1.

if ((tenure1=4 or tenure1=5) and (rent2=1 or rent2=2)) tenharm=2.

if ((tenure1=4 or tenure1=5) and (rent2=3 or rent2=4 or rent2=5 or rent2=6 or rent2=7)) tenharm=3.

if (tenure1=6) tenharm=3.

variable labels tenharm 'ONS Harmonised Tenure type'.

value labels tenharm

- 1 'Owners'
- 2 'Social rented sector'
- 3 'Private rented sector'.

*VEHICLE OWNERSHIP.

if (car=1 or motorcyc=1) vehowner=1.

variable labels vehowner 'Vehicle Ownership'.

value labels vehowner

- 1 'Vehicle Owner'.

*If motorcyc=0 & car=0 then vehowner=sysmis by default.

recode ownbike (1=1) (else=0) into bikowner.

variable labels bikowner 'Bike Owner'.

value labels bikowner

- 1 'Bike Owner'.

recode cartot (1=1) (2=2) (3=3) (4 thru 10=4) into numcars.

variable labels numcars 'Number of cars'.

value labels numcars

- 1 'One'
- 2 'Two'
- 3 'Three'
- 4 'Four or more'.

recode numcars (1 thru 4=copy) (else=0) into numcar2.

variable labels numcar2 'Number of cars (4 categories incl none)'.

value labels numcar2

- 0 'None'
- 1 'One'
- 2 'Two'
- 3 'Three'
- 4 'Four or more'.

recode numcars (1 thru 3=copy) (4=3) (else=0) into numcar3.

variable labels numcar3 'Number of cars (3 categories incl none)'.

value labels numcar3

- 0 'None'
- 1 'One'
- 2 'Two'
- 3 'Three or more'.

recode numcars (1 thru 3=copy) (4=3) into numcar4.

variable labels numcar4 'Number of cars (3 categories)'.

value labels numcar4

1 'One'
2 'Two'
3 'Three or more'.

recode numcars (1 thru 4=1) (else=0) into numcar5.

variable labels numcar5 'Number of cars (2 categories incl none)'.

value labels numcar5

0 'None'
1 'One or more'.

*VEHICLE SECURITY - INCLUDING MAIN CARS.

If ((centloc0=1) or (centloc1=1)) centloc=1.

If ((centloc0=2) or (centloc1=2)) centloc=2.

If ((caralar0=1) or (caralar1=1)) caralar=1.

If ((caralar0=2) or (caralar1=2)) caralar=2.

If ((lmmob0=1) or (lmmob1=1)) lmmobm=1.

If ((lmmob0=2) or (lmmob1=2)) lmmobm=2.

If ((lmmob01=1) or (lmmob11=1)) lmmobe=1.

If ((lmmob01=2) or (lmmob11=2)) lmmobe=2.

If ((lmmobm=1) or (lmmobe=1)) anyimob=1.

If ((lmmobm=2) and (lmmobe=2)) anyimob=2.

If ((vtrack=1) or (vtracka=1)) vtracker=1.

If ((vtrack=2) or (vtracka=2)) vtracker=2.

If ((vetch=1) or (vetch2c=1)) vetchd=1.

If ((vetch=2) or (vetch2c=2)) vetchd=2.

If ((rcascd0=1) or (rcascd1=1)) rcascd=1.

If ((rcascd0=2) or (rcascd1=2)) rcascd=2.

If ((remove0=1) or (remove1=1)) vremove=1.

If ((remove0=2) or (remove1=2)) vremove=2.

If ((secpin0=1) or (secpin1=1)) secpin=1.

If ((secpin0=2) or (secpin1=2)) secpin=2.

If ((vremove=1) or (secpin=1)) radsec=1.

If ((vremove=2) and (secpin=2)) radsec=2.

variable labels centloc 'Central locking'.

caralar 'Car alarm'

lmmobm 'Mechanical immobiliser'

lmmobe 'Electronic immobiliser'

anyimob 'Any immobiliser'

vtracker 'Vehicle tracking device'

vetchd 'Window etching'

rcascd 'Car has radio or cassette or CD player'

vremove 'Removable stereo'

secpin 'Stereo security PIN'

radsec 'Any stereo security'.

Value labels centloc to radsec

1 'Yes'
2 'No'.

if parkdoo0=1 unlocked=1.

```
if parkdoo1=1 unlocked=1.
if parkdoo0=2 unlocked=2.
if parkdoo1=2 unlocked=2.
if parkdoo0=3 unlocked=3.
if parkdoo1=3 unlocked=3.
if parkdoo0=4 unlocked=4.
if parkdoo1=4 unlocked=4.
if parkdoo0=5 unlocked=5.
if parkdoo1=5 unlocked=5.
```

```
if parkdo00=1 notimmob=1.
if parkdo02=1 notimmob=1.
if parkdo00=2 notimmob=2.
if parkdo02=2 notimmob=2.
if parkdo00=3 notimmob=3.
if parkdo02=3 notimmob=3.
if parkdo00=4 notimmob=4.
if parkdo02=4 notimmob=4.
if parkdo00=5 notimmob=5.
if parkdo02=5 notimmob=5.
```

variable labels unlocked 'How often car doors left unlocked'
notimmob 'How often mechanical immobiliser used'.

value labels unlocked notimmob

```
1 'Always'
2 'Almost always'
3 'Sometimes'
4 'Rarely'
5 'Never'.
```

*NUMBER OF ADULTS AND CHILDREN.

recode nadults nchil (1=1) (2=2) (3=3) (4=4) (5 thru 10=5) into nadultgp nchilgrp.

variable labels nadultgp 'Number of adults in household (grouped)'.

variable labels nchilgrp 'Number of children in household (grouped)'.

value labels nadultgp nchilgrp

```
1 'One'
2 'Two'
3 'Three'
4 'Four'
5 'Five or more'.
```

recode nchil (0=0) (1=1) (2=2) (3=3) (4=4) (5 thru 10=5) into nchilgp2.

variable labels nchilgp2 'Number of children in household incl none (grouped)'.

value labels nchilgp2

```
0 'No children'
1 'One'
2 'Two'
3 'Three'
4 'Four'
5 'Five or more'.
```

*HRP AGE.

mHRPick VarRoot = Age NVar = NAdults .

variable labels hrpage 'Age of Household Reference Person'.

recode hrpage (16 thru 29=1)(30 thru 59=2)(60 thru 99=3)(else=sysmis) into hrpagegp.

variable labels hrpagegp 'Age of Household Reference Person (3 bands)'.

value labels hrpagegp

```
1 '16-29'
```

2 '30-59'
3 '60+'.

recode hrpage (16 thru 19=1)(20 thru 24=2)(25 thru 34=3)(35 thru 44 =4)(45 thru 54=5)
(55 thru 64=6)(65 thru 74=7)(75 thru 84=8)(85 thru 99=9)(else=sysmis) into hpageIng.
variable labels hpageIng 'Age of Household Reference Person (9 bands)'.
value labels hpageIng

1 '16-19'
2 '20-24'
3 '25-34'
4 '35-44'
5 '45-54'
6 '55-64'
7 '65-74'
8 '75-84'
9 '85+'.

recode hrpage (16 thru 24=1)(25 thru 44 =2)(45 thru 64=3)(65 thru 74=4)(75 thru 99=5)(else=sysmis) into
hpagesht.

variable labels hpagesht 'Age of Household Reference Person (5 bands)'.
value labels hpagesht

1 '16-24'
2 '25-44'
3 '45-64'
4 '65-74'
5 '75+'.

recode hrpage (16 thru 24=1) (25 thru 34 =2) (35 thru 44 = 3) (45 thru 54 = 4) (55 thru 64 = 5) (65 thru 74 =
6) (75 thru 99 = 7) (else=sysmis) into hrpage7.

variable labels hrpage7 'Age of Household Reference Person (7 bands)'.
value labels hrpage7

1 '16-24'
2 '25-34'
3 '35-44'
4 '45-54'
5 '55-64'
6 '65-74'
7 '75+'.

*HRP OF WORKING AGE.

recode hrpage (16 thru 59=1) into wrkage2.
if ((hrpsex=2) and (wrkage2=1)) hworkage=1.
recode hrpage (16 thru 64=1) into wrkage2.
if ((hrpsex=1) and (wrkage2=1)) hworkage=1.
variable labels hworkage 'Household reference person of working age'.
value labels hworkage
1 'Household reference person of working age'.

*HRP SEX.

mHRPick VarRoot = Sex NVar = NAdults .

variable labels hrpsex 'Sex of Household Reference Person'.
value labels hrpsex
1 'Male'
2 'Female'.

*HRP MARITAL STATUS.

mHRPick VarRoot = Marst NVar = NAdults .

Compute HRPMar = HRPMarst.

```
variable labels hrpmar 'Marital status of Household Reference Person'.
value labels hrpmar
  1 'Single'
  2 'Married and living with'
  3 'Separated'
  4 'Divorced'
  5 'Widowed'.
```

```
mHRPick VarRoot = Cohab NVar = NAdults .
```

```
variable labels hrpcohab 'Cohabiting status of Household Reference Person'.
value labels hrpcohab
  1 'Yes'
  2 'No'
  3 'SPONTANEOUS ONLY - Same sex couple'.
```

```
recode HRPMar (1=3) (2=1) (3=6) (4=5) (5=4) (else = Sysmis) into HRPMarit .
if (hrpcohab=1 or hrpcohab=3) hrpmarit=2.
```

```
variable labels hrpmarit 'Marital status of Household Reference Person'.
value labels hrpmarit
  1 'Married'
  2 'Cohabiting'
  3 'Single'
  4 'Widowed'
  5 'Divorced'
  6 'Separated'.
```

```
*HRP OCCUPATION CODING.
```

```
recode hrpsec2 (1.1=1) (1.2=1) (2=1) (3=2) (4=3) (5=4) (6 thru 7=5) (8=6) (else=sysmis) into hrpssec5.
variable labels hrpssec5 'Occupation coding of HRP (5 categories)'.
```

```
value labels hrpssec5
  1 'Managerial and professional occs'
  2 'Intermediate occs'
  3 'Small employers and own account workers'
  4 'Lower supervisory and technical occs'
  5 'Semi-routine and routine occs'
  6 'Never worked and long term unemployed'.
```

```
recode hrpsec2 (1.1=1) (1.2=1) (2=1) (3=2) (4=3) (5=4) (6 thru 7=5) (8=6) (9=7) (else=sysmis) into hrpssec6.
/* JH */
```

```
variable labels hrpssec6 'Occupation coding of HRP (5 categories expanded)'.
```

```
value labels hrpssec6
  1 'Managerial and professional occs'
  2 'Intermediate occs'
  3 'Small employers and own account workers'
  4 'Lower supervisory and technical occs'
  5 'Semi-routine and routine occs'
  6 'Never worked and long term unemployed'
  7 'Not classified'.
```

```
recode hrpsec (1.0,2.0,3.1,3.2,3.3,3.4,4.1,4.2,4.3,4.4,5.0,6.0=1) (7.1,7.2,7.3,7.4=2) (8.1,8.2,9.1,9.2=3)
(10.0,11.1,11.2=4) (12.1,12.2,12.3,12.4,12.5,12.6,12.7,13.1,13.2,13.3,13.4,13.5=5) (14.1,14.2=6) (15.0=7)
(16.0,17.0=8)
(else=sysmis) into hrpssec6a.
```

```
variable labels hrpssec6a 'Occupation coding of HRP (6 categories & students & not classified)'.
```

```
value labels hrpssec6a
  1 'Managerial and professional occs'
  2 'Intermediate occs'
  3 'Small employers and own account workers'
  4 'Lower supervisory and technical occs'
  5 'Semi-routine and routine occs'
```

6 'Never worked and long term unemployed'
7 'Full-time students'
8 'Not classified'.

recode hrpsec2 (1.1=1) (1.2=1) (2=1) (3 THRU 4=2) (5 THRU 7=3) (8=4) (else=sysmis) into hrpssec3.
variable labels hrpssec3 'Occupation coding of HRP (3 categories)'.

value labels hrpssec3

1 'Managerial and professional occs'
2 'Intermediate occs'
3 'Routine and manual occs'
4 'Never worked and long term unemployed'.

recode hrpsec2 (1.1=1) (1.2=1) (2=1) (3 THRU 4=2) (5 THRU 7=3) (8=4) (9=5) (else=sysmis) into hrpssec4.
variable labels hrpssec4 'Occupation coding of HRP (3 categories expanded)'.

value labels hrpssec4

1 'Managerial and professional occs'
2 'Intermediate occs'
3 'Routine and manual occs'
4 'Never worked and long term unemployed'
5 'Not classified'.

recode hrpsec (1.0,2.0,3.1,3.2,3.3,3.4,4.1,4.2,4.3,4.4,5.0,6.0=1) (7.1,7.2,7.3,7.4,8.1,8.2,9.1,9.2=2)
(10.0,11.1,11.2,12.1,12.2,12.3,12.4,12.5,12.6,12.7,13.1,13.2,13.3,13.4,13.5=3) (14.1,14.2=4) (15.0=5)
(16.0,17.0=6) (else=sysmis) into hrpssec4a.

variable labels hrpssec4a 'Occupation coding of HRP (3 categories & students & not classified)'.

value labels hrpssec4a

1 'Managerial and professional occs'
2 'Intermediate occs'
3 'Routine and manual occs'
4 'Never worked and long term unemployed'
5 'Full-time students'
6 'Not classified'.

*HOUSEHOLD STRUCTURE.

if ((hrpagegp=1 or hrpagegp=2) and (nchil=0)) struct3=1.
if ((hrpagegp=1 or hrpagegp=2) and (nchil gt 0)) struct3=2.
if ((hrpagegp=1 or hrpagegp=2) and (nchil gt 0) and (nadults=1)) struct3=3.
if (hrpagegp=3) struct3=4.
variable labels struct3 'Structure of household'.
value labels struct3
1 'No children'
2 'Children'
3 'Lone parent'
4 'HRP aged 60 plus'.

*SOCIO-ECONOMIC EMPLOYMENT VARIABLES OF RESPONDENT AND HRP (2001 AND 2000
QUESTIONS ON ECONOMIC STATUS ARE VERY DIFFERENT SO DERIVED VARIABLES MAY NOT BE
100% IDENTICAL. FOR EACH OF THE QUESTIONNAIRE VARIABLES THERE IS AN 'R' VERSION FOR
THE RESPONDENT AND 'H' VERSION FOR THE HRP).

*WORK/WORKHRP.

compute rwork = work.
if sysmis(work) rwork=0.
recode rwork (0,8,9=SYSMIS).
variable labels rwork 'Respondent working in last 7 days'.
value labels rwork 1 'Yes' 2 'No'.

if (hrp=1) hwork=work.
if (hrp=2) hwork=workhrp.
if sysmis(work) hwork=0.
recode hwork (SYSMIS=2) (0,8,9=SYSMIS).

variable labels hwork 'HRP working in last 7 days'.
value labels hwork 1 'Yes' 2 'No'.

*GOVTSCH/HGOVTSC.

compute rgvtsch = govtsch.
if sysmis(work) rgvtsch=0.
recode rgvtsch (SYSMIS=2) (0,8,9=SYSMIS).
variable labels rgvtsch 'Respondent on government training scheme'.
value labels rgvtsch 1 'Yes' 2 'No'.

if (hrp=1) hgvtsch=govtsch.
if (hrp=2) hgvtsch=hgovtsch.
if sysmis(work) hgvtsch=0.
recode hgvtsch (SYSMIS=2) (0,8,9=SYSMIS).
variable labels hgvtsch 'HRP on government training scheme'.
value labels hgvtsch 1 'Yes' 2 'No'.

*JOBAWY/HJOBAWY.

compute rjbaway = jobawy.
if sysmis(work) rjbaway=0.
recode rjbaway (SYSMIS=2) (0,8,9=SYSMIS).
variable labels rjbaway 'Respondent away from job'.
value labels rjbaway
1 'Yes'
2 'No'
3 'Waiting to take up job'.

if (hrp=1) hjbaway=jobawy.
if (hrp=2) hjbaway=hjobawy.
if sysmis(work) hjbaway=0.
recode hjbaway (SYSMIS=2) (0,8,9=SYSMIS).
variable labels hjbaway 'HRP away from job'.
value labels hjbaway
1 'Yes'
2 'No'
3 'Waiting to take up job'.

*OWNBUS/HOWNBUS.

compute rowbus=ownbus.
if sysmis(work) rowbus=0.
recode rowbus (SYSMIS=2) (0,8,9=SYSMIS).
variable labels rowbus 'Respondent did unpaid work for own business'.
value labels rowbus 1 'Yes' 2 'No'.

if (hrp=1) howbus=ownbus.
if (hrp=2) howbus=hownbus.
if sysmis(work) howbus=0.
recode howbus (SYSMIS=2) (0,8,9=SYSMIS).
variable labels howbus 'HRP did unpaid work for own business'.
value labels howbus 1 'Yes' 2 'No'.

*RELBUS/HRELBUS.

compute rrlbus=relbus.
if sysmis(work) rrlbus=0.
recode rrlbus (SYSMIS=2) (0,8,9=SYSMIS).
variable labels rrlbus 'Respondent did unpaid work for family business'.
value labels rrlbus 1 'Yes' 2 'No'.

if (hrp=1) hrlbus=relbus.


```

if (hrp=2) hrlbus=hrelbus.
if sysmis(work) hrlbus=0.
recode hrlbus (SYSMIS=2) (0,8,9=SYSMIS).
variable labels hrlbus 'HRP did unpaid work for family business'.
value labels hrlbus 1 'Yes' 2 'No'.

```

*LOOKWK4/HLOOKWK.

```

compute rlkwork=lookwk4.
if sysmis(work) rlkwork=0.
recode rlkwork (SYSMIS=2) (0,8,9=SYSMIS).
variable labels rlkwork 'Respondent was looking for work in last 4 weeks'.
value labels rlkwork 1 'Yes' 2 'No'.

```

```

if (hrp=1) hlkwork=lookwk4.
if (hrp=2) hlkwork=hlookwk.
if sysmis(work) hlkwork=0.
recode hlkwork (SYSMIS=2) (0,8,9=SYSMIS).
variable labels hlkwork 'HRP looking for work in last 4 weeks'.
value labels hlkwork 1 'Yes' 2 'No'.

```

*COMPOSITE VARIABLE OF ECONOMIC STATUS.

```

if sysmis(work) rlstweek=0.
if (rwork=1) rlstweek=1.
if (rgvtsch=1) rlstweek=2.
if (rjbaway=1) rlstweek=3.
if (rowbus=1) or (rrlbus=1) rlstweek=4.
if (rlkwork=1) rlstweek=5.
if (Any(WhyNlook, 1,2,3,4,5,6)) Rlstweek = 5 + WhyNlook.
if (whynlook=8 or whynlook=9) rlstweek=0.
recode rlstweek (0=sysmis).
variable labels rlstweek 'Respondent economic status in last week'.
value labels rlstweek
  1 'Paid work'
  2 'Government training scheme'
  3 'Away from job/waiting to take job up'
  4 'Unpaid work'
  5 'Looking for work'
  6 'Student'
  7 'Looking after family/home'
  8 'Temporarily sick/ill'
  9 'Long-term sick/ill'
  10 'Retired'
  11 'Other'.

```

```

if sysmis(work) hlstweek=0.
if (hrp=1) hlstweek=rlstweek.
if (hrp=2) and (hwork=1) hlstweek=1.
if (hrp=2) and (hgvtsch=1) hlstweek=2.
if (hrp=2) and (hjbaway=1) hlstweek=3.
if (hrp=2) and ((howbus=1) or (hrlbus=1)) hlstweek=4.
if (hrp=2) and (hlkwork=1) hlstweek=5.
if (hrp=2) and (hwhynlk=1) hlstweek=6.
if (hrp=2) and (hwhynlk=2) hlstweek=7.
if (hrp=2) and (hwhynlk=3) hlstweek=8.
if (hrp=2) and (hwhynlk=4) hlstweek=9.
if (hrp=2) and (hwhynlk=5) hlstweek=10.
if (hrp=2) and (hwhynlk=6) hlstweek=11.
if (hrp=2) and (hwhynlk=8 or hwhynlk=9) hlstweek=0.
recode hlstweek (0=sysmis).
variable labels hlstweek 'HRP economic status in last week'.
value labels hlstweek

```

- 1 'Paid work'
- 2 'Government training scheme'
- 3 'Away from job/waiting to take job up'
- 4 'Unpaid work'
- 5 'Looking for work'
- 6 'Student'
- 7 'Looking after family/home'
- 8 'Temporarily sick/ill'
- 9 'Long-term sick/ill'
- 10 'Retired'
- 11 'Other'.

*FULL-TIME STUDENT STATUS.

```
compute rstudy = infstudy.
if (whynlook=1) rstudy=1.
if sysmis(work) rstudy=0.
recode rstudy (sysmis=2) (0,8,9=sysmis).
variable labels rstudy 'Whether respondent full-time student'.
value labels rstudy
  1 'Yes'
  2 'No'.
```

```
if (hrp=1) hstudy=infstudy.
if (hrp=1) and (whynlook=1) hstudy=1.
if (hrp=2) hstudy=hrpstudy.
if (hrp=2) and (hwhynlk=1) hstudy=1.
if sysmis(work) hstudy=0.
recode hstudy (sysmis=2) (0,8,9=sysmis).
variable labels hstudy 'Whether HRP full-time student'.
value labels hstudy
  1 'Yes'
  2 'No'.
```

*EMPLOYMENT.

```
recode RLstWeek (1,2,3,4=1) (5=2) (6,7,8,9,10,11=3) (else=sysmis) into remploy .
variable labels remploy 'Respondent employment status'.
value labels remploy
  1 'Employed'
  2 'Unemployed'
  3 'Economically inactive'.
```

```
recode RLstWeek (1,2,3,4=1) (5=2) (6=3) (7=4) (8,9=5) (10=6) (11=7) (else=sysmis) into remploy2 .
variable labels remploy2 'Respondent employment status (incl inactive breakdowns)'.
value labels remploy2
  1 'Employed'
  2 'Unemployed'
  3 'Economically inactive: student'
  4 'Economically inactive: looking after family/home'
  5 'Economically inactive: long-term/temp sick/ill'
  6 'Economically inactive: retired'
  7 'Economically inactive: other'.
```

```
recode HLstWeek (1,2,3,4=1) (5=2) (6,7,8,9,10,11=3) (else=sysmis) into hemploy .
variable labels hemploy 'HRP employment status'.
value labels hemploy
  1 'Employed'
  2 'Unemployed'
  3 'Economically inactive'.
```

```
recode HLstWeek (1,2,3,4=1) (5=2) (6=3) (7=4) (8,9=5) (10=6) (11=7) (else=sysmis) into hemploy2 .
variable labels hemploy2 'HRP employment status (incl inactive breakdowns)'.
```

value labels hemploy2
1 'Employed'
2 'Unemployed'
3 'Economically inactive: student'
4 'Economically inactive: looking after family/home'
5 'Economically inactive: long-term/temp sick/ill'
6 'Economically inactive: retired'
7 'Economically inactive: other'.

*EVER WORKED.

compute reverw = jobever.
if sysmis(work) reverw=0.
recode reverw (SYSMIS=1) (0,8,9=SYSMIS).
variable labels reverw 'Whether respondent ever had a paid job'.
value labels reverw 1 'Yes' 2 'No'.

if (hrp=1) heverw=jobever.
if (hrp=2) heverw=hrpeverw.
if sysmis(work) heverw=0.
recode heverw (SYSMIS=1) (0,8,9=SYSMIS).
variable labels heverw 'Whether Household Reference Person ever had a paid job'.
value labels heverw 1 'Yes' 2 'No'.

*EMPLOYMENT CHARACTERISTICS OF THOSE WHO DID PAID OR UNPAID WORK LAST WEEK.

*FULL-TIME OR PART TIME WORK.

if any (rlstweek,1,2,3,4) rftpt = ftpt.
recode rftpt (8,9=SYSMIS).
variable labels rftpt 'Respondent working full-time or part-time'.
value labels rftpt
1 'Full-time'
2 'Part-time'.

if (hrp=1) and any (hlstweek,1,2,3,4) hftprt = ftpt.
if (hrp=2) and any (hlstweek,1,2,3,4) hftprt =hftpt.
recode hftprt (8,9=SYSMIS).
variable labels hftprt 'HRP working full-time or part-time'.
value labels hftprt
1 'Full-time'
2 'Part-time'.

*SELF-EMPLOYED OR EMPLOYEE.

if any (rlstweek,1,2,3,4) rselfemp = selfemp.
recode rselfemp (8,9=SYSMIS).
variable labels rselfemp 'Respondent working as employee or self-employed'.
value labels rselfemp
1 'Employee'
2 'Self-employed'.

if (hrp=1) and any (hlstweek,1,2,3,4) hselfemp = selfemp.
if (hrp=2) and any (hlstweek,1,2,3,4) hselfemp =hrpself.
recode hselfemp (8,9=SYSMIS).
variable labels hselfemp 'HRP working as employee or self-employed'.
value labels hselfemp
1 'Employee'
2 'Self-employed'.

*EMPLOYEE STATUS.

if any (rlstweek,1,2,3,4) rmpstat = manage.

```
recode rmpstat (8,9=SYSMIS).
variable labels rmpstat 'Respondent managerial status'.
value labels rmpstat
  1 'Manager'
  2 'Foreman/supervisor'
  3 'Not manager/supervisor'.
```

```
if (hrp=1) and any (hlstweek,1,2,3,4) hmpstat = manage.
if (hrp=2) and any (hlstweek,1,2,3,4) hmpstat =hmanage.
recode hmpstat (8,9=SYSMIS).
variable labels hmpstat 'HRP managerial status'.
value labels hmpstat
  1 'Manager'
  2 'Foreman/supervisor'
  3 'Not manager/supervisor'.
```

*NUMBER OF EMPLOYEES AT PLACE OF WORK.

*This variable has changed from Jun 06 due to change of variable names.

```
if any (rlstweek,1,2,3,4) remplee= nemple2.
if (remplee=1 or remplee=2) remplee=1.
if (remplee=3 or remplee=4) remplee=2.
if (remplee=5 or remplee=6) remplee=3.
if (remplee=9) remplee=0.
recode remplee (0,8=SYSMIS).
variable labels remplee 'Number of employees at respondent"s place of work'.
value labels remplee
  1'Less than 25'
  2'25-499'
  3'Over 500'.
```

```
if (hrp=1) and (any (hlstweek,1,2,3,4)) hemplee= remplee.
if (hrp=2) and (hnemple2=1or hnemple2=2) and (any (hlstweek,1,2,3,4)) hemplee=1.
if (hrp=2) and (hnemple2=3 or hnemple2=4) and (any (hlstweek,1,2,3,4)) hemplee=2.
if (hrp=2) and (hnemple2=5 or hnemple2=6) and (any (hlstweek,1,2,3,4)) hemplee=3.
if (hrp=2) and (hnemple2=9) and (any (hlstweek,1,2,3,4)) hemplee=0.
if (hrp=2) and (hnemple2=8) and (any (hlstweek,1,2,3,4)) hemplee=8.
recode hemplee (0,8=SYSMIS).
variable labels hemplee 'Number of employees at HRP"s place of work'.
value labels hemplee
  1'Less than 25'
  2'25-499'
  3'Over 500'.
```

*SELF-EMPLOYED - WITH OR WITHOUT EMPLOYEES.

*This variable has changed from Jun 06 due to change of variable names.

```
if any(rlstweek,1,2,3,4) remp100=snemp2.
if any(remp100,2,3,4,5,6,7,9) remp100=2.
recode remp100 (8=SYSMIS).
variable labels remp100 'Whether respondent employs people or not'.
value labels remp100
  1'No employees'
  2'With employees'.
```

```
if (hrp=1) and (any(hlstweek,1,2,3,4)) hemp100=remp100.
if (hrp=2) and (any(hlstweek,1,2,3,4)) hemp100=hsnemp2.
if (hrp=2) and (any(hemp100,2,3,4,5,6,7,9)) hemp100=2.
recode hemp100 (8=SYSMIS).
variable labels hemp100 'Whether HRP employs people or not'.
value labels hemp100
  1'No employees'
  2'With employees'.
```

*SELF-EMPLOYED - NUMBER OF PEOPLE EMPLOYED.

*This variable has changed from Jun 06 due to change of variable names.

if any (rlstweek,1,2,3,4) rnemp= snemp2.

if (rnemp=1) rnemp=0.

if (rnemp=2 or rnemp=3) rnemp=1.

if (rnemp=4 or rnemp=5) rnemp=2.

if (rnemp=6 or rnemp=7) rnemp=3.

if (rnemp=9) rnemp=0.

recode rnemp (0,8=SYSMIS).

variable labels rnemp 'Number of people employed by respondent'.

value labels rnemp

1'Less than 25'

2'25-499'

3'Over 500'.

if (hrp=1) and (any (hlstweek,1,2,3,4)) hnemp= rnemp.

if (hrp=2) and (hsnemp2=1) and (any (hlstweek,1,2,3,4)) hnemp=0.

if (hrp=2) and (hsnemp2=2 or hsnemp2=3) and (any (hlstweek,1,2,3,4)) hnemp=1.

if (hrp=2) and (hsnemp2=4 or hsnemp2=5) and (any (hlstweek,1,2,3,4)) hnemp=2.

if (hrp=2) and (hsnemp2=6 or hsnemp2=7) and (any (hlstweek,1,2,3,4)) hnemp=3.

if (hrp=2) and (hsnemp2=9) and (any (hlstweek,1,2,3,4)) hnemp=0.

if (hrp=2) and (hsnemp2=8) and (any (hlstweek,1,2,3,4)) hnemp=8.

recode hnemp (0,8=SYSMIS).

variable labels hnemp 'Number of people employed by HRP'.

value labels hnemp

1'Less than 25'

2'25-499'

3'Over 500'.

*ACCOMMODATION TYPE.

*Creating accharm1.

if (acctyp=1) accharm1=1.

if (acctyp=2) accharm1=2.

if (acctyp=3 OR acctyp=4) accharm1=3.

if (acctyp=5) accharm1=4.

if (acctyp=6) accharm1=5.

if (acctyp=7 OR acctyp=8) accharm1=6.

if (acctyp=9 OR acctyp=10) accharm1=7.

variable labels accharm1 'ONS harmonised accomodation type (7 categories)'.

value labels accharm1

1 'Detached house'

2 'Semi-detached house'

3 'Terraced house'

4 'Maisonette'

5 'Purpose built flat'

6 'Converted flat'

7 'Other accommodation'.

*Creating accharm2.

if any (acctyp, 1, 2, 3, 4) accharm2=1.

if any (acctyp, 5, 6, 7, 8) accharm2=2.

if (acctyp=10 or acctyp=9) accharm2=3.

variable labels accharm2 'ONS harmonised accommodation type (3 categories)'.

value labels accharm2

1 'House'

2 'Flat/maisonette/bedsit'

3 'Other accommodation'.

*Creating accharm3.

if (accharm1=1) accharm3=1.

```

if (accharm1=2) accharm3=2.
if (accharm1=3) accharm3=3.
if (accharm1=4 or accharm1=5 or accharm1=6) accharm3=4.
if (accharm1=7) accharm3=5.
variable labels accharm3 'ONS harmonised accomodation type (5 categories)'.
value labels accharm3
  1 'Detached house'
  2 'Semi-detached house'
  3 'Terraced house'
  4 'Flats/maisonettes'
  5 'Other accommodation'.

```

***INCIVILITY SCALE FROM THE INTERVIEWER POINT OF VIEW.**

```

recode rubbcomm (1 thru 2=1) (3 thru 4=0) (else=sysmis) into rubbish2.
recode vandcomm (1 thru 2=1) (3 thru 4=0) (else=sysmis) into vandals2.
recode poorhou (1 thru 2=1) (3 thru 4=0) (else=sysmis) into poorhou2.

```

```

compute incscale = SUM.3(vandals2,rubbish2, poorhou2) .
variable labels incscale 'Interviewer-rated incivility scale'.

```

```

recode incscale (0 thru 1=0) (2 thru 3=1) into incivgrp.
variable labels incivgrp 'Interviewer-rated incivility scale (grouped)'.
value labels incivgrp
  0 'Low'
  1 'High'.

```

***INCIVILITY SCALE FROM THE RESPONDENT'S POINT OF VIEW.**

```

recode teenhang (1 =2) (2=1) (3 thru 4=0) (else=sysmis) into teen3.
recode vandals (1 =2) (2=1) (3 thru 4=0) (else=sysmis) into vandal3.
recode druguse (1 =2) (2=1) (3 thru 4=0) (else=sysmis) into druguse3.

```

```

compute incscal1 = SUM.3(teen3, vandal3, druguse3) .
variable labels incscal1 'Respondent-rated incivility scale'.

```

```

recode incscal1 (0 thru 3=0) (4 thru 6=1) into resinciv.
variable labels resinciv 'Respondent-rated incivility scale (grouped)'.
value labels resinciv
  0 'Low'
  1 'High'.

```

***PFA - LONDON/NOT LONDON.**

```

recode pfa (1=1) (2 thru 42=2) (else=sysmis) into indlon.
variable labels indlon 'London or outside London (pfa)'.
value labels indlon
  1 'London'
  2 'Outside London'.

```

*** GROUPED GOR.**

```

recode gor (1,2,3=1) (4,5=2) (8,9=3) (6=4) (7=5) (10=6) (else = Sysmis) into gor6 .
variable labels gor6 'Government Office Region (Grouped)'.
value labels gor6
  1 'North'
  2 'Midlands'
  3 'South'
  4 'East'
  5 'London'
  6 'Wales'.

```

```

recode gor (1,2,3,4,5,6,7,8,9=1) (else=sysmis) into engtot.

```

variable labels engtot 'Total for England (gor)'.
value labels engtot
1 'Total for england'.

*AREA TYPE - URBAN/RURAL.

recode rural (1=1) (5=1) (2 thru 4=2) (6 thru 8=2) (else=sysmis) into rural2.
variable labels rural2 'Type of area 2004: urban/rural'.
value labels rural2
1 'Urban'
2 'Rural'.

*HOURS RESPONDENT AWAY FROM HOME ON AVERAGE WEEKDAY.

recode weekday (1,2,3=1) (4=2) (5,6=3) (else=sysmis) into outday.
variable labels outday 'Number of hours respondent away from home on weekday (3-5 hrs)'.
value labels outday
1 'Less than 3hrs'
2 '3-5 hrs'
3 '5 plus hrs'.

recode weekday (1,2,3=1) (4,5=2) (6=3) into weekday2.
variable labels weekday2 'Number of hours respondent away from home on weekday (3-7 hrs)'.
value labels weekday2
1 'Less than 3 hrs'
2 '3-7 hours'
3 '7 plus hrs'.

recode weekday (1 thru 6=copy) (else=sysmis) into weekday3.
variable labels weekday3 'Number of hours respondent away from home on weekday (all categories)'.
value labels weekday3
1 'None'
2 'Less than 1hr'
3 '1-3 hrs'
4 '3-5 hrs'
5 '5-7 hrs'
6 '7 plus hrs'.

*HOURS HOME UNOCCUPIED ON AVERAGE WEEKDAY.

recode unoccl (1,2,3=2) (4=3) (5,6=4) (else=sysmis) into unocccday.
if (weekday=1 or unoccl=2) unocccday=1.
variable labels unocccday 'Number of hours home left unoccupied on average (3-5 hrs incl never)'.
value labels unocccday
1 'Never'
2 'Less than 3 hrs'
3 '3-5 hrs'
4 '5 plus hrs'.

*NO. VISITS PUB/WINE BAR IN EVENING DURING LAST MONTH.

recode pubeve (1=1) (2 thru 3=2) (4 thru 5=3) (else=sysmis) into pub2.
variable labels pub2 'No. visits pub/wine bar evening last month'.
value labels pub2
1 'None'
2 'Less than three times a week'
3 'About three times a week or more often'.

recode pubeve (1=1) (2=2) (3 thru 5=3) (else=sysmis) into pub3.
variable labels pub3 'No. visits pub/wine bar evening last month'.
value labels pub3
1 'None'
2 'Less than once a week'

3 'Once a week or more often'.

*QUALITY OF LIFE AFFECTED BY FEAR OF CRIME.

recode quallife (1 thru 10=copy) (else=sysmis) into quality.
variable labels quality 'Quality of life affected by fear of crime - cleaned variable'.

recode quallife (1 thru 3=1) (4 thru 7=2) (8 thru 10=3) into qualgrp.
variable labels qualgrp 'Quality of life affected by fear of crime (grouped)'.
value labels qualgrp
1 'Minimal'
2 'Moderate'
3 'Great'.

*QUALITY OF LIFE AFFECTED BY CRIME.

recode qualif2 (1 thru 10=copy) (else=sysmis) into quality2.
variable labels quality2 'Quality of life affected by crime - cleaned variable'.

recode qualif2 (1 thru 3=1) (4 thru 7=2) (8 thru 10=3) into qualgrp2.
variable labels qualgrp2 'Quality of life affected by crime (grouped)'.
value labels qualgrp2
1 'Minimal'
2 'Moderate'
3 'Great'.

*CRIME RATE - NATIONAL.

recode crimuk (1 thru 5 =copy) (else =sysmis) into crimuk1a.
variable labels crimuk1a 'Has level of crime in the country changed in the last 2 years - cleaned'.
value labels crimuk1a
1 'A lot more'
2 'A little more'
3 'About the same'
4 'A little less'
5 'A lot less'.

recode crimuk1a (1,2=1) (3 thru 5=2) into crimuk2.
variable labels crimuk2 'Has level of crime in the country changed in the last 2 years (2 categories)'.
value labels crimuk2
1 'More (a lot/a little)'
2 'Less (a lot/a little/about same)'.

*CRIME RATE - LOCAL.

recode crimerat (1 thru 5 =copy) (else =sysmis) into crime1a.
variable labels crime1a 'Has level of crime locally changed in the last 2 years - cleaned'.
value labels crime1a
1 'A lot more'
2 'A little more'
3 'About the same'
4 'A little less'
5 'A lot less'.

recode crime1a (1,2=1) (3 thru 5=2) into crime2.
variable labels crime2 'Has level of crime locally changed in the last 2 years (2 categories)'.
value labels crime2
1 'More (a lot/a little)'
2 'Less (a lot/a little/about same)'.

*HOUSEHOLD SECURITY MEASURES (FROM NVF).

recode burgalar (1=1) (2=0) into burgx.


```
variable labels burgx 'Whether household has burglar alarm'.
value labels burgx
0 'No'
1 'Yes'.
```

```
recode ndummy (1=1) (2=0) into ndummy2.
if (burgalar=1) ndummy2=0.
variable labels ndummy2 'Whether household has dummy alarm'.
value labels ndummy2
0 'No'
1 'Yes'.
```

```
recode visible (1=1) (2=0) into visibl2.
variable labels ndummy2 'Whether household has visible alarm box'.
value labels visibl2
0 'No'
1 'Yes'.
```

```
recode deadlock (1,2=1) (3=0) into deadx.
variable labels deadx 'Whether household has double locks/deadlocks'.
value labels deadx
0 'None'
1 'Yes - on some or all'.
```

```
recode timliout (1=1) (2=0) into timox.
variable labels timox 'Whether household has outdoor lights on timer'.
value labels timox
0 'No'
1 'Yes'.
```

```
recode timliin (1=1) (2=0) into timix.
variable labels timix 'Whether household has indoor lights on timer'.
value labels timix
0 'No'
1 'Yes'.
```

```
if (timliin=2 or timliout=2) lights=0.
if (timliin=1 or timliout=1) lights=1.
variable labels lights 'Whether household has indoor/outdoor lights on timer'.
value labels lights
0 'No'
1 'Yes'.
```

```
recode windlock (1,2,3=1) (4=0) into windlx.
variable labels windlx 'Whether household has window locks'.
value labels windlx
0 'None'
1 'Yes - on some or all'.
```

```
recode bargrill (1,2=1) (3=0) into barx.
variable labels barx 'Whether household has window bars/grilles'.
value labels barx
0 'None'
1 'Yes - on some or all'.
```

```
recode bardoor (1,2,3=1) (4=0) into bardoorx.
variable labels bardoorx 'Whether household has door bars/grilles'.
value labels bardoorx
0 'None'
1 'Yes - at front and/or back'.
```

```
recode chains (1,2=1) (3=0) into chainx.
variable labels chainx 'Whether household has door chains/bars'.
```

value labels chainx
0 'None'
1 'Yes - on some or all'.

recode viewer (1,2=1) (3,4=0) into viewer1.
variable labels viewer1 'Whether household has security viewer in doors'.
value labels viewer1
0 'None'
1 'Yes - on some or all'.

recode viewer (1,2,3=1) (4=0) into viewer2.
variable labels viewer2 'Whether household has security viewer in doors'.
value labels viewer2
0 'None'
1 'Yes - on some or all, or windows'.

*RECODING WORRY ABOUT CRIME.

*BVPI 121 - BURGLARY (RECODED).
recode wburgl (1=2) (2=1) (3=0) (4=0) (else=sysmis) into bvpiburg2 .
variable labels bvpiburg2 "BVPI 121 worry about being burgled - recoded".
value labels bvpiburg2
2 'Very worried'
1 'Fairly worried'
0 'Not very/not at all'.

*BVPI 121 - CAR CRIME (RECODED).
recode wcarstol wfromcar (1=2) (2=1) (3 thru 4=0) (else=sysmis) into wcarstol3 wfromcar3.
compute vehsum3=(wcarstol3 + wfromcar3).
compute newcar=0.
if (car=1) newcar=1.
recode newcar (1=1) (else=sysmis).

*If no car, newcar has been set to missing so bvpicar will be missing if no car.

compute bvpicar2=vehsum3*newcar.
recode bvpicar2 (1 thru 2 = 1) (3 thru 4 = 2).
variable labels bvpicar2 "BVPI 121 worry about car crime - recoded".
value labels bvpicar2
2 'Very worried'
1 'Fairly worried'
0 'Not very/not at all'.

*BVPI 121 -VIOLENT CRIME (RECODED).

recode wmugged wraped wattack wraceatt (1=2) (2=1) (3=0) (4=0) (else=sysmis) into wmugged2 wraped2 wattack2 wraceatt2.
variable labels wmugged2 'How worried about being mugged and robbed - recoded'
wraped2 'How worried about being raped - recoded'
wattack2 'How worried about being physically attacked by strangers - recoded'
wraceatt2 'How worried about being attacked because of skin colour - recoded'.
value labels wmugged2 wraped2 wattack2 wraceatt2
2 'Very worried'
1 'Fairly worried'
0 'Not worried'.

compute bvpiviol2=(wmugged2 + wraped2 + wattack2 + wraceatt2).

recode bvpiviol2 (1 thru 3 = 1) (4 thru 8 = 2).
variable labels bvpiviol2 "BVPI 121 worry about violent crime - recoded".
value labels bvpiviol2
2 'Very worried'
1 'Fairly worried'

0 'Not worried'.

*Add an indicator for the Youngs Persons group, 16 - 24.

RECODE AgeCalib (4=1) (5=1) (SYSMIS=SYSMIS) (ELSE=0) INTO YPSample .

VARIABLE LABELS YPSample 'Young Person, aged 16 to 24'.

Value Labels YPSample 0 "Other Ages" 1 "16-24".

*Compute the number of persons in the Household.

Compute HHMcount = nadults + nchil.

Var Labels hhmcount "Number of members of the household".

*Derive Incidence and Prevalence variables for events in the target period

*Computing two new overall variables which enable an estimate of variance for total crime number estimates (to produce CSEs and confidence intervals around total crime estimate.

*The variables should not be used to estimate total crime, however, as individually they severely over/underestimate crime.

compute allcrmha = (totalhh/nadults) + totperls.

compute allcrmah = (totperls*nadults) + totalhh.

variable labels allcrmha 'all crimes based on converting households into adults'.

variable labels allcrmah 'all crimes based on converting adults into households'.

recode vandalis to violnoi2 allcrmah allcrmha (sysmis=0).

* This syntax derives new incidence variables : multiplies the number of incidents by 10,000 to give the incidence rates.

compute vandal_i=(vandalis*10000).

compute mv.van_i=(mv.vand*10000).

compute homeva_i=(homevand*10000).

compute burgla_i=(burglar*10000).

compute burgat_i=(burgatts*10000).

compute burgno_i=(burgatno*10000).

compute burgen_i=(burgentr*10000).

compute burglo_i=(burgloss*10000).

compute theftd_i=(theftdwe*10000).

compute theftf_i=(theftfmv*10000).

compute thefto_i=(theftomv*10000).

compute attmvt_i=(attmvthf*10000).

compute allmvt_i=(allmvthf*10000).

compute allmvc_i=(allmvcri*10000).

compute biketh_i=(biketthf*10000).

compute othhhc_i=(othhhctd*10000).

compute tohhcl_i=(tohhcltd*10000).

compute totalh_i=(totalhh*10000).

compute acquis_i=(acquisit*10000).

compute hhacq_i=(hhldacq*10000).

Variable labels vandal_i 'Vandalism 80,81,82,83,84,85,86, incidence in last 12 months'

mv.van_i 'Motor vehicle vandalism 81 & 82, incidence in last 12 months'

homeva_i 'Other vandalism 80 83,84,85,86, incidence in last 12 months'

burgla_i 'Burglary 51,52,53, incidence in last 12 months'

burgat_i 'Burglary attempts 53, incidence in last 12 months'

burgno_i 'Burglary attempts and no loss 51 & 53, incidence in last 12 months'

burgen_i 'Burglary with entry 51 & 52, incidence in last 12 months'

burglo_i 'Burglary with loss 52, incidence in last 12 months'

theftd_i 'Theft from a dwelling 55, incidence in last 12 months'

theftf_i 'Theft from a motor vehicle 61 & 63, incidence in last 12 months'

thefto_i 'Theft of a motor vehicle 60 & 62, incidence in last 12 months'

attmvt_i 'Attempted theft of & from vehicle 71 & 72, incidence in last 12 months'

allmvt_i 'All vehicle thefts 60,61,62,63,71 & 72, incidence in last 12 months'

allmvc_i 'All vehicle crime 60,61,62,63,71,72,81,82, incidence in last 12 months'

biketh_i 'Bicycle theft 64, incidence in last 12 months'

othhhc_i 'Other household thefts 50,55,56,57,58,65, incidence in last 12 months'
 tohhcl_i 'Comparable household, 51,52,53,60,61,62,63,64,71,72,80,81,82,83,84,85,86, incidence in last 12 months'
 totalh_i 'All household offences 50,51,52,53,55,56,57,58,60,61,62,63,64,65,71,72,80,81,82,83,84,85,86, incidence in last 12 months'
 acquis_i 'Acquisitive crimes, 43,44,45,51,52,53,60,61,62,63,64,71,72, incidence in last 12 months'
 hhacq_i 'Acquisitive crime against household 50,51,52,53,55,56,57,58,60,61,62,63,64,65,71,72, incidence in last 12 months'.

compute sexoff_i =(sexoffen*10000).
 compute common_i =(commonas*10000).
 compute wound_i =(wounding*10000).
 compute robber_i =(robbery*10000).
 compute theftp_i =(theftper*10000).
 compute thfp.r_i =(thfp.rob*10000).
 compute compvi_i =(compvio*10000).
 compute othpth_i =(othpthef*10000).
 compute topthc_i =(topthcls*10000).
 compute totalp_i =(totalper*10000).
 compute totper_i =(totperls*10000).
 compute allass_i =(allassau*10000).
 compute violen_i =(violence*10000).
 compute threat_i =(threats*10000).
 compute totalb_i =(totalbcs*10000).
 compute alviol_i =(allviol*10000).
 compute mugg1_i =(mugging1*10000).
 compute stealt_i =(stealth*10000).
 compute snatch_i =(snatch*10000).
 compute mug_st_i =(mug_stra*10000).
 compute dom_ac_i =(dom_acq*10000).
 compute domest_i =(domestic*10000).
 compute mugg2_i =(mugging2*10000).
 compute strang_i =(stranger*10000).
 compute acquai_i =(acquain*10000).
 compute vioinj_i =(violinj*10000).
 compute vionoi_i =(violnoi*10000).
 compute peracq_i =(persacq*10000).
 compute alvalc_i =(allvalc*10000).
 compute woundn_i =(woundnew*10000).
 compute asnoij_i =(assnoinj*10000).
 compute cominj_i =(comasinj*10000).
 compute comnij_i =(comasni*10000).
 compute violrs_i =(violnors*10000).
 compute viols_i =(violnos*10000).
 compute vioin2_i =(violinj2*10000).
 compute viono2_i =(violnoi2*10000).

Variable labels sexoff_i 'Sexual offences 31,34 & 35, incidence in last 12 months'
 common_i 'Common assault 13,21, incidence in last 12 months'
 wound_i 'Wounding 11,12,32,33, incidence in last 12 months'
 robber_i 'Robbery 41 42, incidence in last 12 months'
 theftp_i 'Theft from person 43,44,45, incidence in last 12 months'
 thfp.r_i 'Theft from person & robbery 41,42,43,44,45, incidence in last 12 months'
 compvi_i 'comparative violence 13,21,11,12,32,33,41,42, incidence in last 12 months'
 othpth_i 'Other theft of personal property 67,73, incidence in last 12 months'
 topthc_i 'Comparable personal 11,12,32,33,41,42,43,44,45 , incidence in last 12 months'
 totalp_i 'All personal - including sex offences 11,12,13,21,31,32,33,34,35,41,42,43,44,45,67,73, incidence in last 12 months'
 totper_i 'All personal - not including sex offences 11,12,13,21,32,33,41,42,43,44,45,67,73, incidence in last 12 months'
 allass_i 'All assault 11,12,13,21,32,33, incidence in last 12 months'
 violen_i 'Robbery and wounding 11,12,32,33,41,42, incidence in last 12 months'
 threat_i 'Threats 91,92,93,94, incidence in last 12 months'

totalb_i 'Total BCS crime without sex offences
 11,12,13,21,32,33,41,42,43,44,45,50,51,52,53,55,56,57,58,60,61,62,63,64,65,67,71,72,73,80,81,82,83,84,85,86, incidence in last 12 months'
 alviol_i 'All BCS violence 11,12,13,21,32,33,41,42,43, incidence in last 12 months'
 mugg1_i 'Mugging 41,42,43, incidence in last 12 months'
 stealt_i 'Stealth theft from person 44,45, incidence in last 12 months'
 snatch_i 'Snatch theft from person 43, incidence in last 12 months'
 mug_st_i 'Mugging and stranger violence Violgrp2 or 3, incidence in last 12 months'
 dom_ac_i 'Domestic and acquaintance Violgrp1 or 4, incidence in last 12 months'
 domest_i 'Domestic violence Violgrp1, incidence in last 12 months'
 mugg2_i 'Mugging Violgrp2, incidence in last 12 months'
 strang_i 'Stranger Violgrp3, incidence in last 12 months'
 acquai_i 'Acquaintance Violgrp4, incidence in last 12 months'
 vioinj_i 'Violence with injury, incidence in last 12 months'
 vionoi_i 'Violence without injury, incidence in last 12 months'
 peracq_i 'Acquisitive crime against individual 41,42,43,44,45,67,73, incidence in last 12 months'
 alvalc_i 'Alcohol-related all BCS violence, incidence in last 12 months (excl snatch theft)'
 woundn_i 'Wounding (violence with injury), incidence in last 12 months'
 asnoij_i 'Assault with no injury, incidence in last 12 months'
 cominj_i 'Common assault with injury, incidence in last 12 months'
 comnij_i 'Common assault with no injury, incidence in last 12 months'
 violrs_i 'All BCS violence 11,12,13,21,32,33, incidence in last 12 months (excl robbery & snatch theft)'
 viols_i 'All BCS violence 11,12,13,21,32,33,41,41, incidence in last 12 months (excl snatch theft)'
 vioin2_i 'Violence with injury (excl snatch theft), incidence in last 12 months'
 viono2_i 'Violence without injury (excl snatch theft), incidence in last 12 months'.

compute allcha_i =(allcrmha*10000).
 compute allcah_i =(allcrmah*10000).

variable labels allcha_i 'Incidence of all crimes based on converting households into adults - for CSEs not for estimates'

allcah_i 'Incidence of all crimes based on converting adults into households - for CSEs not for estimates'.

*This syntax derives prevalence variables : multiples the number of incidents by 100 to give the prevalence rates.

recode vandalis (0=0) (sysmis=0) (else=100) into vandal_p.
 recode mv.vand (0=0) (sysmis=0) (else=100) into mv.van_p.
 recode homevand (0=0) (sysmis=0) (else=100) into homeva_p.
 recode burglar (0=0) (sysmis=0) (else=100) into burgla_p.
 recode burgatts (0=0) (sysmis=0) (else=100) into burgat_p.
 recode burgatno (0=0) (sysmis=0) (else=100) into burgno_p.
 recode burgentr (0=0) (sysmis=0) (else=100) into burgen_p.
 recode burgloss (0=0) (sysmis=0) (else=100) into burglo_p.
 recode theftdwe (0=0) (sysmis=0) (else=100) into theftd_p.
 recode theftfmv (0=0) (sysmis=0) (else=100) into theftf_p.
 recode theftomv (0=0) (sysmis=0) (else=100) into thefto_p.
 recode attmvthf (0=0) (sysmis=0) (else=100) into attmvt_p.
 recode allmvthf (0=0) (sysmis=0) (else=100) into allmvt_p.
 recode allmvcvri (0=0) (sysmis=0) (else=100) into allmvc_p.
 recode bikethef (0=0) (sysmis=0) (else=100) into biketh_p.
 recode othhhctd (0=0) (sysmis=0) (else=100) into othhhc_p.
 recode tohhcltd (0=0) (sysmis=0) (else=100) into tohhcl_p.
 recode totalhh (0=0) (sysmis=0) (else=100) into totalh_p.
 recode acquisit (0=0) (sysmis=0) (else=100) into acquis_p.
 recode hhldacq (0=0) (sysmis=0) (else=100) into hhacq_p.

Variable labels vandal_p 'Vandalism 80,81,82,83,84,85,86, prevalence in last 12 months'
 mv.van_p 'Motor vehicle vandalism 81 & 82, prevalence in last 12 months'
 homeva_p 'Other vandalism 80,83,84,85,86, prevalence in last 12 months'
 burgla_p 'Burglary 51,52,53, prevalence in last 12 months'
 burgat_p 'Burglary attempts 53, prevalence in last 12 months'
 burgno_p 'Burglary attempts and no loss 51 & 53, prevalence in last 12 months'

burgen_p 'Burglary with entry 51 & 52, prevalence in last 12 months'
 burglo_p 'Burglary with loss 52, prevalence in last 12 months'
 theftd_p 'Theft from a dwelling 55, prevalence in last 12 months'
 thefft_p 'Theft from a motor vehicle 61 & 63, prevalence in last 12 months'
 thefto_p 'Theft of a motor vehicle 60 & 62, prevalence in last 12 months'
 attmvt_p 'Attempted theft of & from vehicle 71 & 72, prevalence in last 12 months'
 allmvt_p 'All vehicle thefts 60,61,62,63,71 & 72, prevalence in last 12 months'
 allmvc_p 'All vehicle crime 60,61,62,63,71,72,81,82, prevalence in last 12 months'
 biketh_p 'Bicycle theft 64, prevalence in last 12 months'
 othhhc_p 'Other household thefts 50,55,56,57,58,65, prevalence in last 12 months'
 tohhcl_p 'Comparable household, 51,52,53,60,61,62,63,64,71,72,80,81,82,83,84,85,86, prevalence in last 12 months'
 totalh_p 'All household offences 50,51,52,53,55,56,57,58,60,61,62,63,64,65,71,72,80,81,82,83,84,85,86, prevalence in last 12 months'
 acquis_p 'Acquisitive crimes, 43,44,45,51,52,53,60,61,62,63,64,71,72, prevalence in last 12 months'
 hhacq_p 'Acquisitive crime against household 50,51,52,53,55,56,57,58,60,61,62,63,64,65,71,72, prevalence in last 12 months'.

recode sexoffen (0=0) (sysmis=0) (else=100) into sexoff_p.
 recode commonas (0=0) (sysmis=0) (else=100) into common_p.
 recode wounding (0=0) (sysmis=0) (else=100) into wound_p.
 recode robbery (0=0) (sysmis=0) (else=100) into robber_p.
 recode theftper (0=0) (sysmis=0) (else=100) into theftp_p.
 recode thfp.rob (0=0) (sysmis=0) (else=100) into thfp.r_p.
 recode compvio (0=0) (sysmis=0) (else=100) into compvi_p.
 recode othpthef (0=0) (sysmis=0) (else=100) into othpth_p.
 recode topthcls (0=0) (sysmis=0) (else=100) into topthc_p.
 recode totalper (0=0) (sysmis=0) (else=100) into totalp_p.
 recode totperls (0=0) (sysmis=0) (else=100) into totper_p.
 recode allassau (0=0) (sysmis=0) (else=100) into allass_p.
 recode violence (0=0) (sysmis=0) (else=100) into violen_p.
 recode threats (0=0) (sysmis=0) (else=100) into threat_p.
 recode totalbcs (0=0) (sysmis=0) (else=100) into totalb_p.
 recode allviol (0=0) (sysmis=0) (else=100) into alviol_p.
 recode mugging1 (0=0) (sysmis=0) (else=100) into mugg1_p.
 recode stealth (0=0) (sysmis=0) (else=100) into stealt_p.
 recode snatch (0=0) (sysmis=0) (else=100) into snatch_p.
 recode mug_stra (0=0) (sysmis=0) (else=100) into mug_st_p.
 recode dom_acq (0=0) (sysmis=0) (else=100) into dom_ac_p.
 recode domestic (0=0) (sysmis=0) (else=100) into domest_p.
 recode mugging2 (0=0) (sysmis=0) (else=100) into mugg2_p.
 recode stranger (0=0) (sysmis=0) (else=100) into strang_p.
 recode acquaint (0=0) (sysmis=0) (else=100) into acquai_p.
 recode violinj (0=0) (sysmis=0) (else=100) into vioinj_p.
 recode violnoi (0=0) (sysmis=0) (else=100) into vionoi_p.
 recode persacq (0=0) (sysmis=0) (else=100) into peracq_p.
 recode allvalc (0=0) (sysmis=0) (else=100) into alvalc_p.
 recode woundnew (0=0) (sysmis=0) (else=100) into woundn_p.
 recode assnoinj (0=0) (sysmis=0) (else=100) into asnoij_p.
 recode comasinj (0=0) (sysmis=0) (else=100) into cominj_p.
 recode comasni (0=0) (sysmis=0) (else=100) into comnij_p.
 recode violnors (0=0) (sysmis=0) (else=100) into violrs_p.
 recode violnos (0=0) (sysmis=0) (else=100) into viols_p.
 recode violinj2 (0=0) (sysmis=0) (else=100) into voin2_p.
 recode violnoi2 (0=0) (sysmis=0) (else=100) into viono2_p.

Variable labels sexoff_p 'Sexual offences 31 34 & 35, prevalence in last 12 months'
 common_p 'Common assault 13 21, prevalence in last 12 months'
 wound_p 'Wounding 11 12 32 33, prevalence in last 12 months'
 robber_p 'Robbery 41 42, prevalence in last 12 months'
 theftp_p 'Theft from person 43,44, 45, prevalence in last 12 months'
 thfp.r_p 'Theft from person & robbery 41,42,43,44, 45, prevalence in last 12 months'
 compvi_p 'comparative violence 13,21,11,12, 32,33,41,42, prevalence in last 12 months'

othpth_p 'Other theft of personal property 67, 73, prevalence in last 12 months'
 topthc_p 'Comparable personal 11 12 32 33 41,42,43,44,45 , prevalence in last 12 months'
 totalp_p 'All personal - including sex offences 11,12,13,21,31,32,33,34,35,41,42,43,44,45,67,73, prevalence in last 12 months'
 totper_p 'All personal - not including sex offences 11,12,13,21,32,33,41,42,43,44,45,67,73, prevalence in last 12 months'
 allass_p 'All assault 11,12,13,21,32,33, prevalence in last 12 months'
 violen_p 'Robbery and wounding 11,12, 32, 33, 41, 42, prevalence in last 12 months'
 threat_p 'Threats 91,92,93,94, prevalence in last 12 months'
 totalb_p 'Total BCS crime without sex offences 11,12,13,21,32,33,41,42,43,44,45,50,51,52,53,55,56, 57,58,60,61,62,63,64,65,67,71,72,73,80,81,82,83,84,85,86, prevalence in last 12 months'
 alviol_p 'All BCS violence 11 12 13 21 32 33 41,42, 43, prevalence in last 12 months'
 mugg1_p 'Mugging 41,42,43, prevalence in last 12 months'
 stealp_p 'Stealth theft from person 44,45, prevalence in last 12 months'
 snatch_p 'Snatch theft from person 43, prevalence in last 12 months'
 mug_st_p 'Mugging and stranger violence Violgrp2 or 3, prevalence in last 12 months'
 dom_ac_p 'Domestic and acquaintance Violgrp1 or 4, prevalence in last 12 months'
 domest_p 'Domestic violence Violgrp1, prevalence in last 12 months'
 mugg2_p 'Mugging Violgrp2, prevalence in last 12 months'
 strang_p 'Stranger Violgrp3, prevalence in last 12 months'
 acquai_p 'Acquaintance Violgrp4, prevalence in last 12 months'
 vioinj_p 'Violence with injury, prevalence in last 12 months'
 vionoi_p 'Violence without injury, prevalence in last 12 months'
 peracq_p 'Acquisitive crime against individual 41,42,43,44,45,67,73, prevalence in last 12 months'
 alvalc_p 'Alcohol-related all BCS violence, prevalence in last 12 months (excl snatch theft)'
 woundn_p 'Wounding (violence with injury), prevalence in last 12 months'
 asnoi_p 'Assault with no injury, prevalence in last 12 months'
 cominj_p 'Common assault with injury, prevalence in last 12 months'
 comnij_p 'Common assault with no injury, prevalence in last 12 months'
 violrs_p 'All BCS violence 11,12,13,21,32,33, prevalence in last 12 months (excl robbery & snatch theft)'
 viols_p 'All BCS violence 11,12,13,21,32,33,41,42, prevalence in last 12 months (excl snatch theft)'
 vion2_p 'Violence with injury (no snatch theft), prevalence in last 12 months'
 viono2_p 'Violence without injury (no snatch theft), prevalence in last 12 months'.

*This syntax derives variables for repeat (more than once) victimisation.

recode vandalis (0=sysmis) (else=copy) into vandal_r.
 recode mv.vand (0=sysmis) (else=copy) into mv.van_r.
 recode homevand (0=sysmis) (else=copy) into homeva_r.
 recode burglar (0=sysmis) (else=copy) into burgla_r.
 recode burgatts (0=sysmis) (else=copy) into burgat_r.
 recode burgatno (0=sysmis) (else=copy) into burgno_r.
 recode burgentr (0=sysmis) (else=copy) into burgen_r.
 recode burgloss (0=sysmis) (else=copy) into burglo_r.
 recode theftdwe (0=sysmis) (else=copy) into theftd_r.
 recode theftfmv (0=sysmis) (else=copy) into theftf_r.
 recode theftomv (0=sysmis) (else=copy) into thefto_r.
 recode attmvthf (0=sysmis) (else=copy) into attmvt_r.
 recode allmvthf (0=sysmis) (else=copy) into allmvt_r.
 recode allmvcvri (0=sysmis) (else=copy) into allmvc_r.
 recode bikethef (0=sysmis) (else=copy) into biketh_r.
 recode othhhctd (0=sysmis) (else=copy) into othhhc_r.
 recode tohhcltd (0=sysmis) (else=copy) into tohhcl_r.
 recode totalhh (0=sysmis) (else=copy) into totalh_r.
 recode acquisit (0=sysmis) (else=copy) into acquis_r.
 recode hhldacq (0=sysmis) (else=copy) into hacq_r.
 recode sexoffen (0=sysmis) (else=copy) into sexoff_r.
 recode commonas (0=sysmis) (else=copy) into common_r.
 recode wounding (0=sysmis) (else=copy) into wound_r.
 recode robbery (0=sysmis) (else=copy) into robber_r.
 recode theftper (0=sysmis) (else=copy) into theftp_r.
 recode thfp.rob (0=sysmis) (else=copy) into thfp.r_r.
 recode compvio (0=sysmis) (else=copy) into compvi_r.

```

recode othpthef (0=sysmis) (else=copy) into othpth_r.
recode toptchls (0=sysmis) (else=copy) into toptch_r.
recode totalper (0=sysmis) (else=copy) into totalp_r.
recode totperls (0=sysmis) (else=copy) into totper_r.
recode allassau (0=sysmis) (else=copy) into allass_r.
recode violence (0=sysmis) (else=copy) into violen_r.
recode threats (0=sysmis) (else=copy) into threat_r.
recode totalbcs (0=sysmis) (else=copy) into totalb_r.
recode allviol (0=sysmis) (else=copy) into alviol_r.
recode mugging1 (0=sysmis) (else=copy) into mugg1_r.
recode stealth (0=sysmis) (else=copy) into stealt_r.
recode snatch (0=sysmis) (else=copy) into snatch_r.
recode mug_stra (0=sysmis) (else=copy) into mug_st_r.
recode dom_acq (0=sysmis) (else=copy) into dom_ac_r.
recode domestic (0=sysmis) (else=copy) into domest_r.
recode mugging2 (0=sysmis) (else=copy) into mugg2_r.
recode stranger (0=sysmis) (else=copy) into strang_r.
recode acquaint (0=sysmis) (else=copy) into acquai_r.
recode violinj (0=sysmis) (else=copy) into vioinj_r.
recode violnoi (0=sysmis) (else=copy) into vionoi_r.
recode persacq (0=sysmis) (else=copy) into peracq_r.
recode woundnew (0=sysmis) (else=copy) into woundn_r.
recode assnoinj (0=sysmis) (else=copy) into asnoij_r.
recode comasinj (0=sysmis) (else=copy) into cominj_r.
recode comasni (0=sysmis) (else=copy) into comnij_r.
recode violnors (0=sysmis) (else=copy) into violrs_r.
recode violnos (0=sysmis) (else=copy) into viols_r.
recode violinj2 (0=sysmis) (else=copy) into voin2_r.
recode violnoi2 (0=sysmis) (else=copy) into viono2_r.

```

```

do repeat r=vandal_r to viono2_r.
if (r gt 2) r=2.
end repeat.

```

Variable labels vandal_r 'Vandalism 80,81,82,83,84,85,86, victimisation in last 12 months'

mv.van_r 'Motor vehicle vandalism 81 & 82, victimisation in last 12 months'

homeva_r 'Other vandalism 80,83,84,85,86, victimisation in last 12 months'

burgla_r 'Burglary 51,52,53, victimisation in last 12 months'

burgat_r 'Burglary attempts 53, victimisation in last 12 months'

burgno_r 'Burglary attempts and no loss 51 & 53, victimisation in last 12 months'

burgen_r 'Burglary with entry 51 & 52, victimisation in last 12 months'

burglo_r 'Burglary with loss 52, victimisation in last 12 months'

theftd_r 'Theft from a dwelling 55, victimisation in last 12 months'

theftf_r 'Theft from a motor vehicle 61 & 63, victimisation in last 12 months'

thefto_r 'Theft of a motor vehicle 60 & 62, victimisation in last 12 months'

attmvt_r 'Attempted theft of & from vehicle 71 & 72, victimisation in last 12 months'

allmvt_r 'All vehicle thefts 60,61,62,63,71 & 72, victimisation in last 12 months'

allmvc_r 'All vehicle crime 60,61,62,63,71,72,81,82, victimisation in last 12 months'

biketh_r 'Bicycle theft 64, victimisation in last 12 months'

othhhc_r 'Other household thefts 50,55,56,57,58,65, victimisation in last 12 months'

tohhcl_r 'Comparable household, 51,52,53,60,61,62,63,64,71,72,80,81,82,83,84,85,86, victimisation in last 12 months'

totalh_r 'All household offences 50,51,52,53,55,56,57,58,60,61,62,63,64,65,71,72,80,81,82,83,84,85,86, victimisation in last 12 months'

acquis_r 'Acquisitive crimes, 43,44,45,51,52,53,60,61,62,63,64,71,72, victimisation in last 12 months'

hhacq_r 'Acquisitive crime against household 50,51,52,53,55,56,57,58,60,61,62,63,64,65,71,72, victimisation in last 12 months'

sexoff_r 'Sexual offences 31,34 & 35, victimisation in last 12 months'

common_r 'Common assault 13,21, victimisation in last 12 months'

wound_r 'Wounding 11,12,32,33, victimisation in last 12 months'

robber_r 'Robbery 41,42, victimisation in last 12 months'

theftp_r 'Theft from person 43,44,45, victimisation in last 12 months'

thfp.r_r 'Theft from person & robbery 41,42,43,44,45, victimisation in last 12 months'

compvi_r 'comparative violence 13,21,11,12,32,33,41,42, victimisation in last 12 months'
 othpth_r 'Other theft of personal property 67,73, victimisation in last 12 months'
 topthc_r 'Comparable personal 11,12,32,33,41,42,43,44,45, victimisation in last 12 months'
 totalp_r 'All personal - including sex offences 11,12,13,21,31,32,33,34,35,41,42,43,44,45,67,73, victimisation in last 12 months'
 totper_r 'All personal - not including sex offences 11,12,13,21,32,33,41,42,43,44,45,67,73, victimisation in last 12 months'
 allass_r 'All assault 11,12,13,21,32,33, victimisation in last 12 months'
 violen_r 'Robbery and wounding 11,12,32,33,41,42, victimisation in last 12 months'
 threat_r 'Threats 91,92,93,94, victimisation in last 12 months'
 totalb_r 'Total BCS crime without sex offences 11,12,13,21,32,33,41,42,43,44,45,50,51,52,53,55,56,57,58,60,61,62,63,64,65,67,71,72,73,80,81,82,83,84,85,86, victimisation in last 12 months'
 alviol_r 'All BCS violence 11,12,13,21,32,33,41,42,43, victimisation in last 12 months'
 mugg1_r 'Mugging 41,42,43, victimisation in last 12 months'
 stealr_r 'Stealth theft from person 44,45, victimisation in last 12 months'
 snatch_r 'Snatch theft from person 43, victimisation in last 12 months'
 mug_st_r 'Mugging and stranger violence Violgrp2 or 3, victimisation in last 12 months'
 dom_ac_r 'Domestic and acquaintance Violgrp1 or 4, victimisation in last 12 months'
 domest_r 'Domestic violence Violgrp1, victimisation in last 12 months'
 mugg2_r 'Mugging Violgrp2, victimisation in last 12 months'
 strang_r 'Stranger Violgrp3, victimisation in last 12 months'
 acquai_r 'Acquaintance Violgrp4, victimisation in last 12 months'
 vioinj_r 'Violence with injury, victimisation in last 12 months'
 vionoi_r 'Violence without injury, victimisation in last 12 months'
 peracq_r 'Acquisitive crime against individual 41,42,43,44,45,67,73, victimisation in last 12 months'
 woundn_r 'Wounding(violence with injury), victimisation in last 12 months'
 asnoij_r 'Assault with no injury, victimisation in last 12 months'
 cominj_r 'Common assault with injury, victimisation in last 12 months'
 comnij_r 'Common assault with no injury, victimisation in last 12 months'
 violrs_r 'All BCS violence 11,12,13,21,32,33, victimisation in last 12 months (excl robbery & snatch theft)'
 viols_r 'All BCS violence 11,12,13,21,32,33,41,42, victimisation in last 12 months (excl snatch theft)'
 vion2_r 'Violence with injury (no snatch theft), victimisation in last 12 months'
 viono2_r 'Violence without injury (no snatch theft), victimisation in last 12 months'.

value labels vandal_r to viono2_r

- 1 'Once'
- 2 'Twice or more'.

recode vandalis (0=sysmis) (else=copy) into vandal_r2.
 recode mv.vand (0=sysmis) (else=copy) into mv.van_r2.
 recode homevand (0=sysmis) (else=copy) into homeva_r2.
 recode burglar (0=sysmis) (else=copy) into burgla_r2.
 recode burgatts (0=sysmis) (else=copy) into burgat_r2.
 recode burgatno (0=sysmis) (else=copy) into burgno_r2.
 recode burgentr (0=sysmis) (else=copy) into burgen_r2.
 recode burgloss (0=sysmis) (else=copy) into burglo_r2.
 recode theftdwe (0=sysmis) (else=copy) into theftd_r2.
 recode theftfmv (0=sysmis) (else=copy) into theftf_r2.
 recode theftomv (0=sysmis) (else=copy) into thefto_r2.
 recode attmvthf (0=sysmis) (else=copy) into attmvt_r2.
 recode allmvthf (0=sysmis) (else=copy) into allmvt_r2.
 recode allmvcri (0=sysmis) (else=copy) into allmvc_r2.
 recode bikethef (0=sysmis) (else=copy) into biketh_r2.
 recode othhhctd (0=sysmis) (else=copy) into othhhc_r2.
 recode tohhcltd (0=sysmis) (else=copy) into tohhcl_r2.
 recode totalhh (0=sysmis) (else=copy) into totalh_r2.
 recode acquisit (0=sysmis) (else=copy) into acquis_r2.
 recode hhldacq (0=sysmis) (else=copy) into hhacq_r2.
 recode sexoffen (0=sysmis) (else=copy) into sexoff_r2.
 recode commonas (0=sysmis) (else=copy) into common_r2.
 recode wounding (0=sysmis) (else=copy) into wound_r2.
 recode robbery (0=sysmis) (else=copy) into robber_r2.

```

recode theftper (0=sysmis) (else=copy) into theftp_r2.
recode thfp.rob (0=sysmis) (else=copy) into thfp.r_r2.
recode compvio (0=sysmis) (else=copy) into compvi_r2.
recode othpthef (0=sysmis) (else=copy) into othpth_r2.
recode topthcls (0=sysmis) (else=copy) into topthc_r2.
recode totalper (0=sysmis) (else=copy) into totalp_r2.
recode totperls (0=sysmis) (else=copy) into totper_r2.
recode allassau (0=sysmis) (else=copy) into allass_r2.
recode violence (0=sysmis) (else=copy) into violen_r2.
recode threats (0=sysmis) (else=copy) into threat_r2.
recode totalbcs (0=sysmis) (else=copy) into totalb_r2.
recode allviol (0=sysmis) (else=copy) into alviol_r2.
recode mugging1 (0=sysmis) (else=copy) into mugg1_r2.
recode stealth (0=sysmis) (else=copy) into stealt_r2.
recode snatch (0=sysmis) (else=copy) into snatch_r2.
recode mug_stra (0=sysmis) (else=copy) into mug_st_r2.
recode dom_acq (0=sysmis) (else=copy) into dom_ac_r2.
recode domestic (0=sysmis) (else=copy) into domest_r2.
recode mugging2 (0=sysmis) (else=copy) into mugg2_r2.
recode stranger (0=sysmis) (else=copy) into strang_r2.
recode acquaint (0=sysmis) (else=copy) into acquai_r2.
recode violinj (0=sysmis) (else=copy) into violinj_r2.
recode violnoi (0=sysmis) (else=copy) into vionoi_r2.
recode persacq (0=sysmis) (else=copy) into peracq_r2.
recode woundnew (0=sysmis) (else=copy) into woundn_r2.
recode assnoinj (0=sysmis) (else=copy) into asnoij_r2.
recode comasinj (0=sysmis) (else=copy) into cominj_r2.
recode comasni (0=sysmis) (else=copy) into comnij_r2.
recode violnors (0=sysmis) (else=copy) into violrs_r2.
recode violnos (0=sysmis) (else=copy) into viols_r2.
recode violinj2 (0=sysmis) (else=copy) into voin2_r2.
recode violnoi2 (0=sysmis) (else=copy) into viono2_r2.

```

```

do repeat r=vandal_r2 to viono2_r2.
if (r gt 3) r=3.
end repeat.

```

```

Variable labels vandal_r2 'Vandalism 80,81,82,83,84,85,86, victimisation in last 12 months'
mv.van_r2 'Motor vehicle vandalism 81 & 82, victimisation in last 12 months'
homeva_r2 'Other vandalism 80,83,84,85,86, victimisation in last 12 months'
burgla_r2 'Burglary 51,52,53, victimisation in last 12 months'
burgat_r2 'Burglary attempts 53, victimisation in last 12 months'
burgno_r2 'Burglary attempts and no loss 51 & 53, victimisation in last 12 months'
burgen_r2 'Burglary with entry 51 & 52, victimisation in last 12 months'
burglo_r2 'Burglary with loss 52, victimisation in last 12 months'
theftd_r2 'Theft from a dwelling 55, victimisation in last 12 months'
theftf_r2 'Theft from a motor vehicle 61 & 63, victimisation in last 12 months'
thefto_r2 'Theft of a motor vehicle 60 & 62, victimisation in last 12 months'
attnvmt_r2 'Attempted theft of & from vehicle 71 & 72, victimisation in last 12 months'
allmvt_r2 'All vehicle thefts 60,61,62,63,71 & 72, victimisation in last 12 months'
allmvc_r2 'All vehicle crime 60,61,62,63,71,72,81,82, victimisation in last 12 months'
biketh_r2 'Bicycle theft 64, victimisation in last 12 months'
othhhc_r2 'Other household thefts 50,55,56,57,58,65, victimisation in last 12 months'
tohhcl_r2 'Comparable household, 51,52,53,60,61,62,63,64,71,72,80,81,82,83,84,85,86, victimisation in last 12 months'
totalh_r2 'All household offences 50,51,52,53,55,56,57,58,60,61,62,63,64,65,71,72,80,81,82,83,84,85,86, victimisation in last 12 months'
acquis_r2 'Acquisitive crimes, 43,44,45,51,52,53,60,61,62,63,64,71,72, victimisation in last 12 months'
hhacq_r2 'Acquisitive crime against household 50,51,52,53,55,56,57,58,60,61,62,63,64,65,71,72, victimisation in last 12 months'
sexoff_r2 'Sexual offences 31,34 & 35, victimisation in last 12 months'
common_r2 'Common assault 13,21, victimisation in last 12 months'
wound_r2 'Wounding 11,12,32,33, victimisation in last 12 months'

```

robber_r2 'Robbery 41,42, victimisation in last 12 months'
 theftp_r2 'Theft from person 43,44,45, victimisation in last 12 months'
 thfp.r_r2 'Theft from person & robbery 41,42,43,44,45, victimisation in last 12 months'
 compvi_r2 'comparative violence 13,21,11,12,32,33,41,42, victimisation in last 12 months'
 othpth_r2 'Other theft of personal property 67,73, victimisation in last 12 months'
 topthc_r2 'Comparable personal 11,12,32,33,41,42,43,44,45, victimisation in last 12 months'
 totalp_r2 'All personal - including sex offences 11,12,13,21,31,32,33,34,35,41,42,43,44,45,67,73, victimisation in last 12 months'
 totper_r2 'All personal - not including sex offences 11,12,13,21,32,33,41,42,43,44,45,67,73, victimisation in last 12 months'
 allass_r2 'All assault 11,12,13,21,32,33, victimisation in last 12 months'
 violen_r2 'Robbery and wounding 11,12,32,33,41,42, victimisation in last 12 months'
 threat_r2 'Threats 91,92,93,94, victimisation in last 12 months'
 totalb_r2 'Total BCS crime without sex offences 11,12,13,21,32,33,41,42,43,44,45,50,51,52,53,55,56, 57,58,60,61,62,63,64,65,67,71,72,73,80,81,82,83,84,85,86, victimisation in last 12 months'
 alviol_r2 'All BCS violence 11,12,13,21,32,33,41,42,43, victimisation in last 12 months'
 mugg1_r2 'Mugging 41,42,43, victimisation in last 12 months'
 stealt_r2 'Stealth theft from person 44,45, victimisation in last 12 months'
 snatch_r2 'Snatch theft from person 43, victimisation in last 12 months'
 mug_st_r2 'Mugging and stranger violence Violgrp2 or 3, victimisation in last 12 months'
 dom_ac_r2 'Domestic and acquaintance Violgrp1 or 4, victimisation in last 12 months'
 domest_r2 'Domestic violence Violgrp1, victimisation in last 12 months'
 mugg2_r2 'Mugging Violgrp2, victimisation in last 12 months'
 strang_r2 'Stranger Violgrp3, victimisation in last 12 months'
 acquai_r2 'Acquaintance Violgrp4, victimisation in last 12 months'
 vioinj_r2 'Violence with injury, victimisation in last 12 months'
 vionoi_r2 'Violence without injury, victimisation in last 12 months'
 peracq_r2 'Acquisitive crime against individual 41,42,43,44,45,67,73, victimisation in last 12 months'
 woundn_r2 'Wounding(violence with injury), victimisation in last 12 months'
 asnoij_r2 'Assault with no injury, victimisation in last 12 months'
 cominj_r2 'Common assault with injury, victimisation in last 12 months'
 comnij_r2 'Common assault with no injury, victimisation in last 12 months'
 violrs_r2 'All BCS violence 11,12,13,21,32,33, victimisation in last 12 months (excl robbery & snatch theft)'
 viols_r2 'All BCS violence 11,12,13,21,32,33,41,42, victimisation in last 12 months (excl snatch theft)'
 vioin2_r2 'Violence with injury (no snatch theft), victimisation in last 12 months'
 viono2_r2 'Violence without injury (no snatch theft), victimisation in last 12 months'.

value labels vandal_r2 to viono2_r2

1 'Once'
 2 'Twice'
 3 'Three or more'.

*121 - burglary: variable="bvpiburg".

recode wburgl (1=1) (2=0) (3=0) (4=0) (else=sysmis) into bvpiburg .

variable labels bvpiburg "BVPI 121 worry about being burgled".

value labels bvpiburg

1 'Very worried'
 0 'Not very/not at all/fairly worried'.

*121 - car crime: variable = "bvpicar".

recode wcarstol wfromcar (1=2) (2=1) (3 thru 4=0) (else=sysmis) into wcarstol2 wfromcar2.

compute vehsum=(wcarstol2 + wfromcar2).

compute newcar=0.

if (car=1) newcar=1.

recode newcar (1=1) (else=sysmis).

*If no car, newcar has been set to missing so bvpicar will be missing if no car.

compute bvpicar=vehsum*newcar.

variable labels bvpicar "BVPI 121 worry about car crime".

value labels wcarstol2 wfromcar2

2 'Very worried'

1 'Fairly worried'
0 'Not worried'.

* 121 - violent crime: variable = "bvpiviol".

compute bvpiviol=(wmugged2 + wraped2 + wattack2 + wraceatt2).

* 7-STRAND ASB MEASURE.

*Recoding 7 strands to get scores: 3=very big problem to 0=not a problem.

recode abancar noisneig drunk druguse teenhang rubbish vandals (1=3) (2=2) (3=1) (4=0) (else=sysmis)
into
abancar7 nois7 drun7 drug7 teen7 rubb7 vand7 .

compute nasb7 = (abancar7 + nois7 + drun7+ drug7 + teen7 + rubb7 + vand7).
variable labels nasb7 'Perceived level of anti-social behaviour'.

*INDIVIDUAL STRANDS OF DISORDER.

recode teenhang (1=1) (2=1) (3=0) (4=0) (else=sysmis) into teen1.
recode vandals (1=1) (2=1) (3=0) (4=0) (else=sysmis) into vandals1.
recode racehat2 (1=1) (2=1) (3=0) (4=0) (else=sysmis) into race1.
recode druguse (1=1) (2=1) (3=0) (4=0) (else=sysmis) into drug1.
recode drunk (1=1) (2=1) (3=0) (4=0) (else=sysmis) into drunk1.
recode rubbish (1=1) (2=1) (3=0) (4=0) (else=sysmis) into rubb1.
recode noisneig (1=1) (2=1) (3=0) (4=0) (else=sysmis) into noisneg1.
recode abancar (1=1) (2=1) (3=0) (4=0) (else=sysmis) into abancar1.

*CONFIDENCE IN CJS.

recode conffwit (1=1) (2=1) (3=0) (4=0) (else=sysmis) into confwit2 .
recode conffoff (1=1) (2=1) (3=0) (4=0) (else=sysmis) into conffoff2 .
recode confvict (1=1) (2=1) (3=0) (4=0) (else=sysmis) into confvic2 .
recode confrig (1=1) (2=1) (3=0) (4=0) (else=sysmis) into confrig2 .
recode confcase (1=1) (2=1) (3=0) (4=0) (else=sysmis) into confcas2 .

recode effred (1=1) (2=1) (3=0) (4=0) (else=sysmis) into effred2 .
recode effyng (1=1) (2=1) (3=0) (4=0) (else=sysmis) into effyng2 .

*PUBLIC SATISFACTION WITH THE POLICE (HOW GOOD A JOB QUESTION).

recode jobpol (1 thru 2=100) (3 thru 5=0) (else=sysmis) into jobpol1.
variable label jobpol1 'Public satisfaction with the police'.
value labels jobpol1
100 'Excellent or good job'
0 'Fair or poor job'.

*PUBLIC SATISFACTION WITH THE LOCAL POLICE (HOW GOOD A JOB QUESTION).

recode ratpol2 (1 thru 2=100) (3 thru 5=0) (else=sysmis) into ratpol3.
variable label ratpol3 'Public satisfaction with the police'.
value labels ratpol3
100 'Excellent or good job'
0 'Fair or poor job'.

*CRIME RATE IN THE UK.

recode crimuk (1 =100) (2 thru 5 = 0) (else =sysmis) into crimuk1.
variable labels crimuk1 'Has level of crime in the country changed in last 2 years - recoded'.
value labels crimuk1
100 'Lot more'
0 'Not_lot more'.

*CRIME RATE LOCALLY.

```
recode crimerat (1 =100) (2 thru 5 = 0) (else =sysmis) into crime1.  
variable labels crime1 'Has level of crime locally changed in last 2 years - recoded'.  
value labels crime1  
  100 'Lot more'  
   0 'Not_lot more'.
```

*WITNESS SATISFACTION.

```
recode polsatwi (1 thru 4=copy) (else = sysmis) into witsat.  
variable labels witsat 'How satisfied or dissatisfied were you with MOST RECENT contact with the police -  
recoded'.  
value labels witsat  
  1 'Very satisfied'  
  2 'Fairly satisfied'  
  3 'A bit dissatisfied'  
  4 'Very dissatisfied'.
```

```
recode witsat (1,2 = 1) (3,4 = 0) (else = sysmis) into witsat2.
```

*KDI's new policing questions.

```
Recode polatt1 (1 thru 2 = 100) (3 thru 5 = 0) (else=sysmis) into patt1.  
Recode polatt2 (1 thru 2 = 100) (3 thru 5 = 0) (else=sysmis) into patt2.  
Recode polatt3 (1 thru 2 = 100) (3 thru 5 = 0) (else=sysmis) into patt3.  
Recode polatt4 (1 thru 2 = 100) (3 thru 5 = 0) (else=sysmis) into patt4.  
Recode polatt5 (1 thru 2 = 100) (3 thru 5 = 0) (else=sysmis) into patt5.  
Recode polatt6 (1 thru 2 = 100) (3 thru 5 = 0) (else=sysmis) into patt6.  
Recode polatt7 (1 thru 2 = 100) (3 thru 5 = 0) (else=sysmis) into patt7.  
variable labels patt1 'Police in this area can be relied on to be there when you need - recoded'  
patt2 'Police in this area would treat you with respect if you had contact with them - recoded'  
patt3 'Police in this area treat everyone fairly regardless of who they are - recoded'  
patt4 'Police in this area can be relied on to deal with minor crimes - recoded'  
patt5 'Police in this area understand the issues that affect this community - recoded'  
patt6 'Police in this area are dealing with the things that matter to this community - recoded'  
patt7 'Taking everything into account I have confidence in the police in this area - recoded'.  
value labels patt1 to patt7  
  100 'Agree'  
   0 'Neither_Disagree'.
```

*Recoding all variables to (0,100) - police satisfaction variable already coded as 0,100.

```
if (bvpiburg gt 0) bvpiburg=100.  
if (bvpicar lt 3) bvpicar=0.  
if (bvpicar ge 3) bvpicar=100.  
if (bvpiviol lt 4) bvpiviol=0.  
if (bvpiviol ge 4) bvpiviol=100.  
variable labels bvpiburg 'BVPI 121 worry about being burgled'  
bvpicar 'BVPI 121 worry about car crime'  
bvpiviol 'BVPI 121 worry about violent crime'.  
value labels bvpiburg bvpicar bvpiviol  
  100 'High level of worry'  
   0 'Not high level of worry'.
```

```
if (confwit2 gt 0) confwit2=100.  
if (conffoff2 gt 0) conffoff2=100.  
if (confvic2 gt 0) confvic2=100.  
if (confrig2 gt 0) confrig2=100.  
if (confcas2 gt 0) confcas2=100.  
variable labels confwit2 'Witnesses are treated well'  
conffoff2 'Bringing people to justice'  
confvic2 'Meeting the needs of victims'
```

```
confrig2 'Respect rights and treats people fairly'
confcas2 'Deals with cases promptly and efficiently'.
value labels confwit2 confoff2 confvic2 confrig2 confcas2
  100 'Very/fairly confident'
  0 'Not very/Not at all confident'.
```

```
if (effred2 gt 0) effred2=100.
if (effyng2 gt 0) effyng2=100.
variable labels effred2 'Effective in reducing crime'
effyng2 'Effective in dealing with young people'.
value labels effred2 effyng2
  100 'Very/fairly effective'
  0 'Not very/Not at all effective'.
```

```
if (teen1 gt 0) teen1=100.
if (vandals1 gt 0) vandals1=100.
if (race1 gt 0) race1=100.
if (drug1 gt 0) drug1=100.
if (drunk1 gt 0) drunk1=100.
if (rubb1 gt 0) rubb1=100.
if (noisneg1 gt 0) noisneg1=100.
if (abancar1 gt 0) abancar1=100.
variable labels teen1 'How much of a problem are teenagers hanging around'
vandals1 'How much of a problem is vandalism, graffiti, etc'
race1 'How much of a problem is attack because of skin colour'
drug1 'How much of a problem is people using or dealing drugs'
drunk1 'How much of a problem is people being drunk or rowdy'
rubb1 'How much of a problem is litter/rubbish'
noisneg1 'How much of a problem are noisy neighbours'
abancar1 'How much of a problem are abandoned or burnt out cars'.
value labels teen1 vandals1 race1 drug1 drunk1 rubb1 noisneg1 abancar1
  100 'Very/fairly big problem'
  0 'Not very big/ not a problem at all'.
```

```
if (nasb7 lt 11) nasb7hi=0.
if (nasb7 ge 11) nasb7hi=100.
variable labels nasb7hi 'Perceived level of anti-social behaviour - high/low rate'.
value labels nasb7hi
  100 'High overall ASB'
  0 'Not high overall ASB'.
```

```
if (witsat2 gt 0) witsat2=100.
variable labels witsat2 'How satisfied/dissatisfied were you with MOST RECENT police contact (2 groups)'.
value labels witsat2
  100 'Satisfied'
  0 'Dissatisfied'.
```

*HOUSEHOLD SECURITY MEASURES (MERGED DATA FROM VF AND NVF).

```
recode burgalar (1,2=0) into secdeva.
if (burgalar = 1) secdeva = 1.
recode secalarm (1=1) (0=0) into secdeva.
variable labels secdeva 'Whether property has burglar alarm (from VF and NVF)'.
value labels secdeva
  1 'Has burglar alarm'.
```

```
if ((ndummy =1) or (ndummy=2) or (burgalar=1)) secdevb=0.
if (ndummy = 1) secdevb = 1.
recode secdumm (1=1) (0=0) into secdevb.
variable labels secdevb 'Whether property has dummy alarm (from VF and NVF)'.
value labels secdevb
  1 'Has dummy alarm'.
```

```

recode deadlock (1,2,3=0) into secdevc.
if ((deadlock = 1) or (deadlock=2)) secdevc = 1.
recode secdead (1=1) (0=0) into secdevc.
variable labels secdevc 'Whether property has deadlocks (from VF and NVF)'.
value labels secdevc
  1 'Has deadlocks'.

```

```

recode timliout (1,2=0) into secdevd.
if (timliout = 1) secdevd = 1.
recode secoutlt (1=1) (0=0) into secdevd.
variable labels secdevd 'Whether property has outside lights (from VF and NVF)'.
value labels secdevd
  1 'Has outside lights'.

```

```

recode timliin (1,2=0) into secdeve.
if (timliin = 1) secdeve = 1.
recode secinlit (1=1) (0=0) into secdeve.
variable labels secdeve 'Whether property has inside security lights (from VF and NVF)'.
value labels secdeve
  1 'Has inside security lights'.

```

```

recode windlock (1,2,3,4=0) into secdevf.
if ((windlock = 1) or (windlock=2) or (windlock=3)) secdevf = 1.
recode secwin (1=1) (0=0) into secdevf.
variable labels secdevf 'Whether property has window locks (from VF and NVF)'.
value labels secdevf
  1 'Has window locks'.

```

```

recode bargrill (1,2,3=0) into secdevg.
if ((bargrill = 1) or (bargrill=2)) secdevg = 1.
recode secwbar (1=1) (0=0) into secdevg.
variable labels secdevg 'Whether property has bars or grills on windows (from VF and NVF)'.
value labels secdevg
  1 'Has bars or grills on windows'.

```

```

recode chains (1,2,3=0) into secdevh.
if ((chains = 1) OR (chains=2)) secdevh = 1.
recode secchain (1=1) (0=0) into secdevh.
variable labels secdevh 'Whether property has door chain (from VF and NVF)'.
value labels secdevh
  1 'Has door chain'.

```

```

*Excluding dummy alarms.
compute seccount=(secdeva+secdevc+secdevd+secdeve+secdevf+secdevg+secdevh).
variable labels seccount 'Number of home security devices (from VF and NVF)'.

```

```

recode seccount (0 thru 4 =copy) (5 thru 7=5) into secnum.
variable labels secnum 'Grouped number of home security devices (from VF and NVF)'.
value labels secnum
  1 'One security device'
  2 'Two security devices'
  3 'Three security devices'
  4 'Four security devices'
  5 'Five or more security devices'.

```

*CREATING HIGH, LOW AND NO SECURITY VARIABLE.

```

if ((secdeva=1) or (secdeva=0)) highsec=3.
if ((secdevc=1) or (secdevf=1) or (secdevh=1)) highsec=2.
if ((secdeva=1) or (secdevd=1) or (secdeve=1) or (secdevg=1)) highsec=1.
variable labels highsec 'Level of home security'.
value labels highsec
  1 'High security'

```

2 'Low security'

3 'No security'.

APPENDIX E

Home Office derived variables for the 2006/07 BCS victim form datafile

*RECODING OFFENCES.

*TYPE OF BURGLARY.

recode offence (52=1) (51,53=2) into burg1.
variable labels burg1 'Type of burglary - loss or no loss'.
value labels burg1
1 'Burglary with loss'
2 'Attempted burglary or no loss'.

recode offence (53=1) (51,52=2) into burg2.
variable labels burg2 'Type of burglary - entry or attempt'.
value labels burg2
1 'Attempted burglary'
2 'Burglary with entry'.

*TYPE OF CRIMINAL DAMAGE.

recode offence (80, 83, 84, 85, 86=1) (81, 82=2) into VanHomVe.
variable labels VanHomVe 'Type of criminal damage - home or vehicle'.
value labels VanHomVe
1 'Criminal damage to home and property around home'
2 'Criminal damage to vehicle'.

recode vanhomve (1=1) (2=0) into filthome.
recode vanhomve (1=0) (2=1) into filtveh.
variable labels filthome 'Filter for criminal damage to the home'.
variable labels filtveh 'Filter for criminal damage to vehicles'.

*TYPE OF THEFT.

recode offence (50,57=0) (55=1) (56=2) (58=3) (65=4) into thefthhd.
variable labels thefthhd 'Type of theft'.
value labels thefthhd
0 'Attempt/burglary to nonconnected garage/outhouse - nothing taken'
1 'Theft in a dwelling'
2 'Theft from a meter'
3 'Burglary to nonconnected garage or outhouse - something taken'
4 'Theft outside dwelling'.

Recode offence (43=1) (44=2) (45=3) into pertheft.
Variable labels pertheft 'Type of theft from person'.
Value labels pertheft
1 'Snatch theft from person'
2 'Stealth theft from person'
3 'Attempted theft from the person'.

*TYPE OF CAR THEFT.

If (offence = 60) cartheft = 1.
If (offence = 62) cartheft = 2.
variable labels cartheft 'Type of car theft - car/van or motorbike/moped'.
value labels cartheft
1 'Theft of car/van'
2 'Theft of motorbike/moped'.

if (offence=60 or offence=62) vehtheft=1.
if (offence=61 or offence=63) vehtheft=2.
If (offence=71 or offence=72) vehtheft=3.
variable labels vehtheft 'Vehicle theft of or from or attempts'.

value labels vehtheft
1 'Theft of vehicle'
2 'Theft from vehicle'
3 'Attempted theft of/from vehicle'.

if (vehtheft=1 or vehtheft =2) vehtht2= 1.
If (vehtheft=3) vehtht2=2.
variable labels vehtht2 'Vehicle theft of or from or attempts'.
value labels vehtht2
1 'Theft of/from vehicle'
2 'Attempted theft of/from vehicle'.

if (vehtheft=1) vehtht3= 1.
If (vehtheft=2) vehtht3=2.
variable labels vehtht3 'Vehicle theft of or from'.
value labels vehtht3
1 'Theft of vehicle'
2 'Theft from vehicle'.

if (offence=71) attcar=1.
variable labels attcar 'Attempted theft of/from car/van'.
value labels attcar
1 'Attempted theft of/from car/van'.

*CRIMINAL DAMAGE TO THE HOME.

Do if ((whatdama=1) or (whatdamb=1) or (whatdamc=1) or (whatdamd=1) or (whatdame=1) or (whatdamf=1) or (whatdamg=1)).
Do repeat r=hmdmoth homewind homedoor homegraf homesoil homeoth wallgraf wallbrke walloth shedwind sheddoor shedgraf shedsoil shedoth.
Compute r=0.
End repeat.
End if.

If ((whatdama=1) or (whatdamb=1) or (whatdamg=1)) hmdmoth=1.
variable labels hmdmoth 'Other criminal damage'.
value labels hmdmoth
1 'Other criminal damage'.

If (damhoma=1) homewind=1.
variable labels homewind 'House/flat - window broken'.
value labels homewind
1 'House/flat - window broken'.

If ((damhomb=1) or (damhomc=1)) homedoor=1.
variable labels homedoor 'House/flat - door/lock damaged'.
value labels homedoor
1 'House/flat - door/lock damaged'.

If (damhomd=1) homegraf=1.
variable labels homegraf 'House/flat - graffiti'.
value labels homegraf
1 'House/flat - graffiti'.

If (damhome=1) homesoil=1.
variable labels homesoil 'House/flat - soiling'.
value labels homesoil
1 'House/flat - soiling'.

If ((damhomf=1) or (damhomg=1) or (damhomh=1)) homeoth=1.
variable labels homeoth 'House/flat - other criminal damage'.
value labels homeoth
1 'House/flat - other criminal damage'.

If (damwalla=1) wallgraf=1.
variable labels wallgraf 'Wall/fence/other garden - graffiti'.
value labels wallgraf
1 'Wall/fence/other garden - graffiti'.

If (damwallb=1) wallbrke=1.
variable labels wallbrke 'Wall/fence/other garden - broken'.
value labels wallbrke
1 'Wall/fence/other garden - broken'.

If ((damwallc=1) or (damwalld=1)) walloth=1.
variable labels walloth 'Wall/fence/other garden - other'.
value labels walloth
1 'Wall/fence/other garden - other'.

If (damgara=1) shedwind=1.
variable labels shedwind 'Shed/garage - window broken'.
value labels shedwind
1 'Shed/garage - window broken'.

If ((damgarb=1) or (damgarc=1)) sheddoor=1.
variable labels sheddoor 'Shed/garage - door/lock damaged'.
value labels sheddoor
1 'Shed/garage - door/lock damaged'.

If (damgard=1) shedgraf=1.
variable labels shedgraf 'Shed/garage - graffiti'.
value labels shedgraf
1 'Shed/garage - graffiti'.

If (damgare=1) shedsoil=1.
variable labels shedsoil 'Shed/garage - soiling'.
value labels shedsoil
1 'Shed/garage - soiling'.

If ((damgarf=1) or (damgarg=1) or (damgarh=1) or (damgari=1) or (damgarj=1)) shedoth=1.
variable labels shedoth 'Shed/garage - other'.
value labels shedoth
1 'Shed/garage - other'.

*TIME OF INCIDENT.

recode timevic2 (1 thru 7 = copy) (else=sysmis) into time1.
variable labels time1 'Time of incident (7 groups)'.
value labels time1
1 'During morning'
2 'During afternoon'
3 'Morning/afternoon'
4 'Early evening'
5 'Late evening'
6 'During night'
7 'Evening/night'.

recode time1 (1 thru 3 =copy) (4 thru 5 = 4) (6=5) (7=6) (else=sysmis) into time2.
variable labels time2 'Time of incident (6 groups)'.
value labels time2
1 'During morning'
2 'During afternoon'
3 'Morning/afternoon'
4 'During evening'
5 'During night'
6 'Evening/night'.

recode time1 (1=1) (2=1) (3=1) (4=2) (5=2) (6=2) (7=2) into time3.
variable labels time3 'Time of incident (2 groups)'.
value labels time3
1 'Day (morning or afternoon)'
2 'Night (evening or night)'.

*WHEN INCIDENT TOOK PLACE.

recode whenvic2 (1 = 1) (2 thru 6 = 2) (else=sysmis) into when1.
variable labels when1 'When incident took place (2 groups)'.
value labels when1
1 'Week'
2 'Weekend'.

*WHERE INCIDENT TOOK PLACE.

recode whervic2 (1 thru 10 = copy) (11=12) (12=11) (13=12) (else=sysmis) into whervic3.
variable labels whervic3 'Where incident took place'.
value labels whervic3
1 'Own home or own garage (including attempted break-in)'
2 'Immediately outside home (inc. shed, garden, street)'
3 'In or near victim"s place of work (including work car park)'
4 'In public car park'
5 'In/around pub/bar/night club/working men"s club'
6 'In/around dancehall/disco'
7 'In/around football ground/other sports ground'
8 'In/around sports centre/sports club'
9 'In/around other place of public entertainment'
10 'Travelling on transport or near transport facilities'
11 'Other public or commercial locations'
12 'Elsewhere'.

*LOCATION OF INCIDENT.

if (ownhome=1) locate1=1.
if (ownhome=2) locate1=2.
if (ownhome=3 or outhome=7) locate1=5.
if (outhome=1) locate1=3.
if (outhome = 3 or outhome = 4) locate1=4.
if (outhome=5) locate1=6.
if (outhome=6)locate1=11.
if (outhome=2) locate1=12.
if (vicwork=1) locate1=7.
if (vicwork=2) locate1=8.
if (vicwork=3) locate1=65.
if (vicwork=4) locate1=9.
if (vicwork=5) locate1=10.
if (pubbar=1) locate1=71.
if (pubbar=2) locate1=72.
if (pubbar=3) locate1=73.
if (pubbar=4 or pubbar=9)locate1=74.
if (disco=1) locate1=13.
if (disco=2)locate1=14.
if (disco=3) locate1=15.
if (disco = 4 or disco = 9) locate1=16.
if (sportgr=1) locate1=17.
if (sportgr=2) locate1=18.
if (sportgr=3)locate1=19.
if (sportgr = 4 or sportgr=9)locate1=20.
if (pubent=1) locate1=21.
if (pubent=2) locate1=22.
if (pubent=3) locate1=23.

if (pubent=4 or pubent = 9)locate1=24.
if (transp=1) locate1=25.
if (railstat=1) locate1=26.
if (railstat=2) locate1=27.
if (railstat=3) locate1=28.
if (railstat = 4 or railstat =9)locate1=29.
if (transp=3)locate1=30.
if (tubestat=1) locate1=31.
if (tubestat=2) locate1=32.
if (tubestat=3) locate1=33.
if (tubestat = 4 or tubestat = 9) locate1=34.
if (transp=5) locate1=35.
if (busstop=1) locate1=36.
if (busstop=2) locate1=37.
if (busstop=3 or busstop = 9) locate1=38.
if (transp=7) locate1=39.
if (airport=1) locate1=40.
if (airport=2) locate1=41.
if (airport = 3 or airport = 9) locate1=42.
if (supmkt=1) locate1=43.
if (supmkt=2) locate1=44.
if (supmkt=3) locate1=45.
if (supmkt = 4 or supmkt = 9) locate1=46.
if (college=1) locate1=47.
if (college=2) locate1=48.
if (college=3) locate1=49.
if (college =4 or college = 9)locate1=50.
if (relfri=1) locate1=55.
if (relfri=2) locate1=56.
if (relfri=3) locate1=57.
if (relfri =4 or relfri = 9) locate1=58.
if (sportcl=1) locate1=75.
if (sportcl=2) locate1=76.
if (sportcl=3) locate1=77.
if (sportcl = 4 or sportcl = 9) locate1=78.
if (mosque=1) locate1=79.
if (mosque=2) locate1=80.
if (mosque=3) locate1=81.
if (mosque = 4 or mosque = 9) locate1=82.
if (comprem=4) locate1=83.
if (comoth=1) locate1=51.
if (comoth=2) locate1=52.
if (comoth=3) locate1=53.
if (comoth = 4 or comoth= 9 or comprem=9) locate1=54.
if (elsewher=2) locate1=59.
if (elsewher=3 or whervic2=4) locate1=60.
if (elsewher=4) locate1=61.
if (elsewher=5) locate1=62.
if (elsewher=6) locate1=63.
if (elsewher=7) locate1=64.
if (elsewher=8) locate1=70.
if (elsewher=9) locate1=66.
if (elsewher=10) locate1=67.
if (elsewher=11) locate1=97.
if (elsewher = 12 or elsewher = 99) locate1=98.
if (ownhome=9 or outhome=9) locate1=68.
if (transp=99) locate1=69.
if (vicwork=9) locate1=84.
if (transp=9) locate1 = 85.
if (transp=10) locate1 =86.
if (whervic2=11) locate1=87.
if (transp=98) locate1=99.
if (vicwork=8) locate1=99.

if (ownhome=8 or outhome=8 or whervic2=98 or elsewhere=98) locate1=99.
if (whervic2=99) locate1=98.

variable labels locate1 'Location of incident'.

value labels locate1

- 1 'Inside your own home'
- 2 'In a garage next to this house/flat'
- 3 'Inside the same building (corridor, stairs, lift, etc.)'
- 4 'In a garden/outside the building on same premises'
- 5 'In another garage (e.g. row of garages for flats)'
- 6 'In the street outside your house/flat'
- 7 'At a place of work - inside building'
- 8 'At a place of work - out of doors'
- 9 'In a car park at a place of work'
- 10 'In the street near a place of work'
- 11 'In a car park for this estate'
- 12 'In a shed, greenhouse, or other outbuilding on premises'
- 13 'Inside dancehall/disco'
- 14 'Car park of dancehall/disco'
- 15 'Street outside dancehall/disco'
- 16 'Not sure where at dancehall/disco'
- 17 'Inside football/sports ground'
- 18 'Car park of football/sports ground'
- 19 'Street outside football/sports ground'
- 20 'Not sure where at football/sports ground'
- 21 'Inside place of public entertainment'
- 22 'Car park of place of public entertainment'
- 23 'Street outside place of public entertainment'
- 24 'Not sure where at place of public entertainment'
- 25 'On a train'
- 26 'Train station: in the station, on a platform'
- 27 'Train station: in a car park'
- 28 'Train station: on the street outside'
- 29 'Train station: not sure where'
- 30 'On a tube train'
- 31 'Tube station: in the station, on a platform'
- 32 'Tube station: in a car park'
- 33 'Tube station: on the street outside'
- 34 'Tube station: not sure where'
- 35 'On a bus'
- 36 'Bus stop: at bus stop, station or street outside'
- 37 'Bus stop: in a car park'
- 38 'Bus stop: not sure where'
- 39 'On a plane'
- 40 'At the airport'
- 41 'In the airport car park'
- 42 'Not sure where at the airport'
- 43 'Inside a supermarket'
- 44 'In a supermarket car park'
- 45 'In the street/shopping precinct outside supermarket'
- 46 'Not sure where at supermarket'
- 47 'Inside college'
- 48 'In a college car park'
- 49 'In the street /in the grounds outside college'
- 50 'Not sure where at college'
- 51 'Inside other commercial premises'
- 52 'In car park of other commercial premises'
- 53 'In the street/in the grounds outside other commercial premises'
- 54 'Not sure where at other commercial premises'
- 55 'Inside a friends/relatives house'
- 56 'In a friends/relatives garage/car port/car park'
- 57 'In the street outside friends/relatives house'
- 58 'Not sure where at friends/relatives house'

59 'Inside another building'
 60 'In another carpark'
 61 'In subway under street or road'
 62 'In another street or road'
 63 'At a park, common, or other public open space'
 64 'At an urban waste ground or building site'
 65 'In a garage at work'
 66 'On a boat'
 67 'At allotments'
 68 'Dont know where at home or outside home'
 69 'Dont know where on transport'
 70 'At a caravan site'
 71 'Inside pub/bar/nightclub'
 72 'Car park of pub/bar/nightclub'
 73 'Street outside pub/bar/nightclub'
 74 'Not sure where at pub/bar/nightclub'
 75 'Inside sports centre/club'
 76 'Car park of sports centre/club'
 77 'Street outside sports centre/club'
 78 'Not sure where at sports centre/club'
 79 'Inside church/mosque'
 80 'Car park of church/mosque'
 81 'Street outside church/mosque'
 82 'Not sure where at church/mosque'
 83 'At a street market'
 84 'Dont know where at victims place of work'
 85 'In a taxi'
 86 'Driving or travelling in a car/van'
 87 'In/around petrol station forecourt'
 97 'Other specific location'
 98 'Dont know or vague answer'
 99 'Not answered' .

*LOCATION OF INCIDENT - RECODED.

recode locate1

(1,2,3,55,56=1)

(4,5,11,12,68=2)

(6=3)

(7,8,9,65,84=4)

(10=5)

(72,14,18,76,22,27,32,37,41,44,48,80,52,60,70=6)

(73,15,19,77,23,28,33,45,49,81,53,57,62,63,64=7)

(71,74,13,16,17,20,75,78,21,24,25,26,29,30,31,34,35,36,38,39,40,42,43,46,47,50,79,82,83,85,86,51,54,58,59,61,67,66,87,69,97=8)

into newloc.

variable labels newloc 'Location of incident (8 groups) original version'.

value labels newloc

1 'Home - private'

2 'Home - semi private'

3 'Home - street'

4 'Work - car park'

5 'Work - street'

6 'Other - car park'

7 'Other street'

8 'Other'.

recode newloc (1 thru 3 =1) (4=2) (6=2) (5=3) (7=3)(8=4) into newloc2.

variable labels newloc2 'Location of incident (4 groups) original version'.

value labels newloc2

1 'Home'

2 'Car park'

3 'Street not home'
4 'Other'.

recode locate1

(1,2,3,55,56=1)

(4,5,11,12=2)

(6=3)

(9,65=4)

(10=5)

(7,8=6)

(72,14,18,76,22,27,32,37,41,44,48,80,52,60=7)

(73,15,19,77,23,28,33,45,49,81,53,57,62=8)

(71,74,13,16,17,20,75,78,21,24,25,26,29,30,31,34,35,36,38,39,40,42,43,46,47,50,79,82,83,85,51,54,61,67,6
6,87,69=9)

(86,58,59,68,84,97=10)

into newloc3.

variable labels newloc3 'Location of incident (10 groups) new version 2006-07'.

value labels newloc3

1 'Home - private'

2 'Home - semi private'

3 'Home - street'

4 'Work - car park'

5 'Work - street'

6 'Work - other'

7 'Other - car park'

8 'Other - street'

9 'Public place'

10 'Other'.

recode locate1

(1,2,3,55,56=1)

(4,5,11,12=2)

(6=3)

(9,65=4)

(10=5)

(7,8=6)

(72,14,18,76,22,27,32,37,41,44,48,80,52,60=7)

(73,15,19,77,23,28,33,45,49,81,53,57,62=8)

(25,26,29,30,31,34,35,36,38,39,40,42,85,66,69=9)

(71,74,13,16,17,20,75,78,21,24,43,46,47,50,79,82,83,51,54,61,67,87=10)

(86,58,59,68,84,97=11)

into newloc4.

variable labels newloc4 'Location of incident (11 groups) new version 2006-07'.

value labels newloc4

1 'Home - private'

2 'Home - semi private'

3 'Home - street'

4 'Work - car park'

5 'Work - street'

6 'Work - other'

7 'Other - car park'

8 'Other - street'

9 'Public transport'

10 'Other public place'

11 'Other'.

if any(locate1,17,18,20,21, 22, 24,

43,44,46,47,48,50,51,52,54,55,56,58,59,60,66,67,70,75,76,78,79,80,82,87,97,98,99) loctype=1.

if any(locate1,1,2,3,4,5,6,11,12,68) loctype=2.

if any(locate1, 7,8,9,65,84) loctype=3.

if any(locate1,10,19,23,28,33,45,49,53,57,61,62,63,64,77,81,83) loctype=4.

if any(locate1,13,14,15,16,71,72,73,74) loctype=5.

if any(locate1,25,26,27,29,30,31,32,34,35,36,37,38,39,40,41,42,69,85,86) loctype=6.
variable labels loctype 'Location of incident (for violent crime)'.
value labels loctype
1 'Other location'
2 'Around the home'
3 'Around work'
4 'Street'
5 'Pub or club'
6 'Public transport'.

*Creating single variables for each type of security.

if ((vcentlo1=1) or (vcentlo2=1)) vcentlo=1.
if ((vcentlo1=2) or (vcentlo2=2)) vcentlo=2.

if ((vcarala1=1) or (vcarala2=1)) vcarala=1.
if ((vcarala1=2) or (vcarala2=2)) vcarala=2.

if ((vimmob1=1) or (vimmob3=1)) vimmobm=1.
if ((vimmob1=2) or (vimmob3=2)) vimmobm=2.

if ((vimmob2=1) or (vimmob4=1)) vimmobe=1.
if ((vimmob2=2) or (vimmob4=2)) vimmobe=2.

if ((vimmobm=1) or (vimmobe=1)) vanyimob=1.
if ((vimmobm=2) and (vimmobe=2)) vanyimob=2.

if ((vvtrack1=1) or (vvtrack2=1)) vvtrack=1.
if ((vvtrack1=2) or (vvtrack2=2)) vvtrack=2.

if ((vvetch1=1) or (vvetch2=1)) vvetch=1.
if ((vvetch1=2) or (vvetch2=2)) vvetch=2.

if ((vrcascd1=1) or (vrcascd2=1)) vrcascd=1.
if ((vrcascd1=2) or (vrcascd2=2)) vrcascd=2.

if ((vremove1=1) or (vremove2=1)) vremove=1.
if ((vremove1=2) or (vremove2=2)) vremove=2.

if ((vsecpin1=1) or (vsecpin2=1)) vsecpin=1.
if ((vsecpin1=2) or (vsecpin2=2)) vsecpin=2.

if ((vremove=1) or (vsecpin=1)) vradsec=1.
if ((vremove=2) and (vsecpin=2)) vradsec=2.

variable labels
vcentlo 'Central locking'
vcarala 'Car alarm'
vimmobm 'Mechanical immobiliser'
vimmobe 'Electronic immobiliser'
vanyimob 'Any immobiliser'
vvtrack 'Vehicle tracking device'
vvetch 'Window etching'
vrcascd 'Car has radio or cassette or CD player'
vremove 'Removable stereo'
vsecpin 'Stereo security PIN'
vradsec 'Any stereo security'.
value labels vcentlo to vradsec
1 'Yes'
2 'No'.

*VEHICLE AGE.

recode vehage (1 thru 3=1) (4=2) into vehage2.
variable labels vehage2 'Grouped age of vehicle'.
value labels vehage2
1 'Up to 10 years old'
2 'More than 10 years or old'.

***TYPE OF CONTACT WITH OFFENDER.**

if (athome=2) contype1=1.
if (athome=1 and aware=2) contype1=2.
if (athome=1 and aware=1 and descroff=2) contype1=3.
if ((athome=1 and aware=1) and (descroff=1 or v78=1)) contype1=4.
variable labels contype1 'Type of contact with offender'.
value labels contype1
1 'Away from home'
2 'At home but unaware'
3 'At home and aware'
4 'At home, aware and saw offender'.

recode descroff (2,9=1) (else=2) into deskroff.
variable labels deskroff 'Description of offenders'.
value labels deskroff
1 'No/Dont know anything about offenders'
2 'Yes - know something about offenders'.

recode athome (1=1) (2=2) (else=sysmis) into athome2.
variable labels athome2 'Whether anyone at home during incident'.
value labels athome2
1 'Someone at home'
2 'No-one at home'.

***POINT OF ENTRY FOR BURGLARY.**

recode frontbac (1 thru 4=copy) (else=sysmis) into frontbac2.
variable labels frontbac2 'Tried to/got in through front or back of property'.

if any (frontbac2, 1, 2, 3, 4) door=0.
if (entdoor=1) door=1.
variable labels door 'Burglary entry: tried to/got in through door'.
value labels door
1 'Tried to/got in through door'.

if any (frontbac2, 1, 2, 3, 4) window=0.
if (throwind=1) window=1.
variable labels window 'Burglary entry: tried to/got in through window'.
value labels window
1 'Tried to/got in through window'.

if any (frontbac2, 1, 2, 3, 4) otherent=0.
if (othentry=1) otherent=1.
variable labels otherent 'Burglary entry: tried to/got in any other way'.
value labels otherent
1 'Tried to/got in any other way'.

***MODE OF ENTRY THROUGH DOOR.**

if (entdoor=1 or entdoor=2) doorpush=0.
if (howdoo2a=1) doorpush=1.
variable labels doorpush 'How offender got through door: pushed past'.
value labels doorpush
1 '(Tried to) push past person who opened door'.

if (entdoor=1 or entdoor=2) doorlock=0.

if (howdoo2b=1) doorlock=1.
variable labels doorlock 'How offender got through door: door not locked'.
value labels doorlock
1 'Door was not locked'.

if (entdoor=1 or entdoor=2) doorkey=0.
if (howdoo2c=1) doorkey=1.
variable labels doorkey 'How offender got through door: they had key'.
value labels doorkey
1 'Offender had key'.

if (entdoor=1 or entdoor=2) doorforc=0.
if (howdoo2d=1) doorforc=1.
variable labels doorforc 'How offender got through door: force/break lock'.
value labels doorforc
1 '(Tried to) force/break lock'.

if (entdoor=1 or entdoor=2) doorcut=0.
if (howdoo2e=1) doorcut=1.
variable labels doorcut 'How offender got through door: break/cut out panel'.
value labels doorcut
1 '(Tried to) break/cut out panel'.

if (entdoor=1 or entdoor=2) doorcut=0.
if (howdoo2f=1) doorpret=1.
variable labels doorpret 'How offender got through door: false pretences'.
value labels doorpret
1 'By false pretences'.

if (entdoor=1 or entdoor=2) doorglas=0.
if (howdoo2g=1) doorglas=1.
variable labels doorglas 'How offender got through door: break/smash glass'.
value labels doorglas
1 '(Tried to) break/smash glass'.

if (entdoor=1 or entdoor=2) doorkick=0.
if (howdoo2h=1) doorkick=1.
variable labels doorkick 'How offender got through door: kick/smash/ram door'.
value labels doorkick
1 '(Tried to) kick/smash/ram door'.

if (entdoor=1 or entdoor=2) dooroth=0.
if (howdoo2i=1) dooroth=1.
variable labels dooroth 'How offender got through door: other'.
value labels dooroth
1 'Other method to get through door'.

*MODE OF ENTRY THROUGH WINDOW.

if (throwind=1 or throwind=2) winpush=0.
if (howwinda=1) winpush=1.
variable labels winpush 'How offender got through window: open'.
value labels winpush
1 'Window was open'.

if (throwind=1 or throwind=2) winlock=0.
if (howwindb=1) winlock=1.
variable labels winlock 'How offender got through window: lock/catch'.
value labels winlock
1 '(Tried to) force window lock/catch'.

if (throwind=1 or throwind=2) wincut=0.
if (howwindc=1) wincut=1.

variable labels wincut 'How offender got through window: break/cut out glass'.
value labels wincut
1 '(Tried to) break/cut out glass'.

if (throwind=1 or throwind=2) winoth=0.
if (howwindd=1) winoth=1.
variable labels winoth 'How offender got through window: other'.
value labels winoth
1 'Other method to get through window'.

***ENTRY BY FALSE PRETENCES.**

if (offinho2=1 or offinho2=2 or tryinsi2=1 or tryinsi2=2) falseent=0.
if (offinvit=3 or insidefp=1 or tryfpins=1 or howdoo2f=1) falseent=1.
variable labels falseent '(Tried to) get in using false pretences'.
value labels falseent
1 '(Tried to) get in using false pretences'.

***ANY DAMAGE TO HOUSE/FLAT.**

compute anydam=0.
if (damhoma=1 or damhomb=1 or damhomc=1 or damhomd=1 or damhome=1 or damhomf=1 or
damhomg=1 or damhomh=1) anydam=1.
variable labels anydam 'Any damage to house/flat'.
value labels anydam
1 'Damage'
0 'No damage'.

***SERIOUSNESS OF CRIME.**

recode scorcrm2 (99=sysmis) (98=sysmis) (else=copy) into score1.
variable labels score1 'Score for seriousness of crime (excl DK/RF)'.
recode score1 (1 thru 6=1) (7 thru 13=2) (14 thru 20=3) (else=sysmis) into score2.
variable labels score2 'Grouped score for seriousness of crime'.
value labels score2
1 '1 to 6'
2 '7 to 13'
3 '14 to 20'.

***USE OF THREAT OR FORCE.**

if (useforce=1 or v710=1) force=1.
if (useforce=2) force=2.
variable labels force 'Whether force or violence used'.
value labels force
1 'Yes - used force'
2 'No - force not used'.

if (threavio=1 or v711=1 or useforce=1 or v710=1) throrfor=1.
if (threavio=2 and useforce=2) throrfor=2.
variable labels throrfor 'Whether force or violence threatened/used'.
value labels throrfor
1 'Yes - threat/use of force/violence'
2 'No - threat/use of force/violence'.

***COST OF DAMAGE.**

***Sets damage or replacement cost to zero if no damage to property.**

do if (defaprop=2 or defaprop=9).
compute totdamgp=0.
compute totdamgp2=0.
compute totdamgp3=0.

end if.

do if (stolitem=2 or stolitem=9).

compute valveh2=0.

compute othvalvh2=0.

compute totvalue2=0.

end if.

recode totdamag (0=0) (1 thru 49=1) (50 thru 99=2) (100 thru 249=3) (250 thru 499=4) (500 thru 999=5) (1000 thru 4999=6) (5000 thru 50000=7) into totdamgp.

variable labels totdamgp 'Grouped cost of damage (7 groups)'.

value labels totdamgp

0 '£0'

1 '£1-£49'

2 '£50-£99'

3 '£100-£249'

4 '£250-£499'

5 '£500-£999'

6 '£1,000-£4,999'

7 '£5,000-£50,000'.

recode totdamag (0=0) (1 thru 49=1) (50 thru 99=2) (100 thru 249=3) (250 thru 499=4) (500 thru 999=5) (1000 thru 4999=6) (5000 thru 99997=7) into totdamgp3.

variable labels totdamgp3 'Grouped cost of damage (7 groups) including maximum'.

value labels totdamgp3

0 '£0'

1 '£1-£49'

2 '£50-£99'

3 '£100-£249'

4 '£250-£499'

5 '£500-£999'

6 '£1,000-£4,999'

7 'More than £5,000'.

recode totdamag valveh othvalvh totvalue (1 thru 49=1) (50 thru 99=2) (100 thru 499=3) (500 thru 999=4) (1000 thru 4999=5) (5000 thru 9999=6)

(10000 thru 14999=7) (15000 thru 99997=8) into totdamgp2 valveh2 othvalvh2 totvalue2.

variable labels

totdamgp2 'Grouped cost of damage (8 groups)'

valveh2 'Grouped replacement cost of stolen vehicle (8 groups)'

othvalvh2 'Grouped value of other stolen property (8 groups)'

totvalue2 'Grouped total replacement cost of what was stolen (8 groups)'.

value labels totdamgp2 valveh2 othvalvh2 totvalue2

0 'No damage/items stolen'

1 '£1-£49'

2 '£50-£99'

3 '£100-£499'

4 '£500-£999'

5 '£1,000-£4,999'

6 '£5,000-£9,999'

7 '£10,000-£14,999'

8 'More than £15,000'.

*ITEMS STOLEN.

*Creating duplicate copy of items stolen variable for recoding.

Compute xwhast2a=whast2a.

Compute xwhast2b=whast2b.

Compute xwhast2c=whast2c.

Compute xwhast2d=whast2d.

Compute xwhast2e=whast2e.

Compute xwhast2f=whast2f.

Compute xwhast2g=whast2g.
Compute xwhast2h=whast2h.
Compute xwhast2i=whast2i.
Compute xwhast2j=whast2j.
Compute xwhast2k=whast2k.
Compute xwhast2l=whast2l.
Compute xwhast2m=whast2m.
Compute xwhast2n=whast2n.
Compute xwhast2o=whast2o.
Compute xwhast2p=whast2p.
Compute xwhast2q=whast2q.
Compute xwhast2r=whast2r.
Compute xwhast2s=whast2s.
Compute xwhast2t=whast2t.
Compute xwhast2u=whast2u.
Compute xwhast2v=whast2v.
Compute xwhast2w=whast2w.
Compute xwhast2x=whast2x.
Compute xwhast2y=whast2y.
Compute xwhast2z=whast2z.
Compute xwhast2aa=whast2aa.
Compute xwhast2bb=whast2bb.
Compute xwhast2cc=whast2cc.
Compute xwhast2dd=whast2dd.
Compute xwhast2ee=whast2ee.
Compute xwhast2ff=whast2ff.
Compute xwhast2gg=whast2gg.
Compute xwhast2hh=whast2hh.
Compute xwhast2ii=whast2ii.
Compute xwhast2jj=whast2jj.
Compute xwhast2kk=whast2kk.
Compute xwhast2ll=whast2ll.
Compute xwhast2mm=whast2mm.
Compute xwhast2nn=whast2nn.
Compute xwhast2oo=whast2oo.
Compute xwhast2pp=whast2pp.
Compute xwhast2qq=whast2qq.

Compute xwhast3a=whast3a.
Compute xwhast3b=whast3b.
Compute xwhast3c=whast3c.
Compute xwhast3d=whast3d.
Compute xwhast3e=whast3e.
Compute xwhast3f=whast3f.
Compute xwhast3g=whast3g.
Compute xwhast3h=whast3h.
Compute xwhast3i=whast3i.
Compute xwhast3j=whast3j.
Compute xwhast3k=whast3k.
Compute xwhast3l=whast3l.
Compute xwhast3m=whast3m.
Compute xwhast3n=whast3n.
Compute xwhast3o=whast3o.
Compute xwhast3p=whast3p.
Compute xwhast3q=whast3q.
Compute xwhast3r=whast3r.
Compute xwhast3s=whast3s.
Compute xwhast3t=whast3t.
Compute xwhast3u=whast3u.
Compute xwhast3v=whast3v.
Compute xwhast3w=whast3w.
Compute xwhast3x=whast3x.
Compute xwhast3y=whast3y.

Compute xwhast3z=whast3z.
 Compute xwhast3aa=whast3aa.
 Compute xwhast3bb=whast3bb.
 Compute xwhast3cc=whast3cc.
 Compute xwhast3dd=whast3dd.
 Compute xwhast3ee=whast3ee.
 Compute xwhast3ff=whast3ff.
 Compute xwhast3gg=whast3gg.
 Compute xwhast3hh=whast3hh.
 Compute xwhast3ii=whast3ii.
 Compute xwhast3jj=whast3jj.
 Compute xwhast3kk=whast3kk.
 Compute xwhast3ll=whast3ll.
 Compute xwhast3mm=whast3mm.
 Compute xwhast3nn=whast3nn.
 Compute xwhast3oo=whast3oo.
 Compute xwhast3pp=whast3pp.
 Compute xwhast3qq=whast3qq.
 Compute xwhast3rr=whast3rr.

*Removing DK and Refused.

Do if ((xwhast2pp=1) or (xwhast2qq=1)).
 Recode xwhast2a to xwhast2oo (1,0=sysmis).
 End if.

Do if ((xwhast3qq=1) or (xwhast3rr=1)).
 Recode xwhast3a to xwhast3pp (1,0=sysmis).
 End if.

*Burglary: combining categories.

Do repeat r=whast_b1 to whast_b21.
 If ((xwhast2a=1) or (xwhast2a=0) or (xwhast3a=1) or (xwhast3a=0)) r=0.
 End repeat.

If ((xwhast2a=1) or (xwhast2b=1) or (xwhast2c=1) or (xwhast3a=1) or (xwhast3b=1) or (xwhast3c=1))
 whast_b1=1.
 If ((xwhast2e=1) or (xwhast3e=1) or (xwhast2f=1) or (xwhast3f=1) or (xwhast2g=1) or (xwhast3g=1) or
 (xwhast2h=1) or (xwhast3h=1)) whast_b2=1.
 If ((xwhast2i=1) or (xwhast3i=1)) whast_b3=1.
 If ((xwhast2j=1) or (xwhast3j=1)) whast_b4=1.
 If ((xwhast2k=1) or (xwhast3k=1)) whast_b5=1.
 If ((xwhast2l=1) or (xwhast3l=1)) whast_b6=1.
 If ((xwhast2m=1) or (xwhast3m=1) or (xwhast2n=1) or (xwhast3n=1) or (xwhast2o=1) or (xwhast3o=1) or
 (xwhast2p=1) or (xwhast3p=1)
 or (xwhast2bb=1) or (xwhast3cc=1) or (xwhast2y=1) or (xwhast3y=1)) whast_b7=1.
 If ((xwhast2q=1) or (xwhast3q=1)) whast_b8=1.
 If ((xwhast2r=1) or (xwhast3r=1)) whast_b9=1.
 If ((xwhast2s=1) or (xwhast3s=1)) whast_b10=1.
 If ((xwhast2t=1) or (xwhast3t=1)) whast_b11=1.
 If ((xwhast2u=1) or (xwhast3u=1) or (xwhast2z=1) or (xwhast3aa=1)) whast_b12=1.
 If ((xwhast2v=1) or (xwhast3v=1) or (xwhast2mm=1) or (xwhast3nn=1)) whast_b13=1.
 If ((xwhast2w=1) or (xwhast3w=1)) whast_b14=1.
 If ((xwhast2x=1) or (xwhast3x=1)) whast_b15=1.
 If ((xwhast2bb=1) or (xwhast2nn=1) or (xwhast3cc=1) or (xwhast3oo=1)) whast_b16=1.
 If ((xwhast2dd=1) or (xwhast3ee=1)) whast_b17=1.
 If ((xwhast2aa=1) or (xwhast2gg=1) or (xwhast2oo=1) or (xwhast2ii=1) or (xwhast3bb=1) or (xwhast3hh=1)
 or (xwhast3pp=1) or (xwhast3jj=1)) whast_b18=1.
 If ((xwhast2cc=1) or (xwhast2jj=1) or (xwhast2kk=1) or (xwhast2ll=1) or (xwhast3dd=1) or (xwhast3kk=1) or
 (xwhast3ll=1) or (xwhast3mm=1)) whast_b19=1.
 If ((xwhast2ee=1) or (xwhast2ff=1) or (xwhast2hh=1) or (xwhast3ff=1) or (xwhast3gg=1) or (xwhast3ii=1))
 whast_b20=1.

If ((xwhast2d=1) or (xwhast3d=1)) whast_b21=1.

Variable labels whast_b1 'Burglary: stolen vehicle/ vehicle parts'
whast_b2 'Burglary: stolen money/wallet/purse/chequebook/credit cards'
whast_b3 'Burglary: stolen mobile phone'
whast_b4 'Burglary: stolen jewellery'
whast_b5 'Burglary: stolen clothes'
whast_b6 'Burglary: stolen documents'
whast_b7 'Burglary: stolen electrical goods/cameras'
whast_b8 'Burglary: stolen computer/computer equipment'
whast_b9 'Burglary: stolen CDs/tapes/videos/DVDs'
whast_b10 'Burglary: stolen house keys'
whast_b11 'Burglary: stolen car keys'
whast_b12 'Burglary: stolen tools or work materials'
whast_b13 'Burglary: stolen bicycle or bicycle parts'
whast_b14 'Burglary: stolen garden furniture'
whast_b15 'Burglary: stolen wheely bin/ dustbin'
whast_b16 'Burglary: stolen childrens toys or baby items'
whast_b17 'Burglary: stolen sports equipment'
whast_b18 'Burglary: other stolen items'
whast_b19 'Burglary: stolen household items/furniture'
whast_b20 'Burglary: stolen food/toiletries/cigarettes'
whast_b21 'Burglary: stolen briefcase/handbag/shopping bag'.

Value labels whast_b1 to whast_b21

0 'No'
1 'Yes'.

*Theft from person: combining categories of items stolen.

*Removing DK and Refused and combining categories.

Do repeat r=whast_t1 to whast_t16.

If ((xwhast2a=1) or (xwhast2a=0) or (xwhast3a=1) or (xwhast3a=0)) r=0.

End repeat.

If ((xwhast2d=1) or (xwhast3d=1)) whast_t1=1.

If ((xwhast2e=1) or (xwhast3e=1)) whast_t2=1.

If ((xwhast2f=1) or (xwhast3f=1)) whast_t3=1.

If ((xwhast2g=1) or (xwhast3g=1)) whast_t4=1.

If ((xwhast2h=1) or (xwhast3h=1)) whast_t5=1.

If ((xwhast2i=1) or (xwhast3i=1)) whast_t6=1.

If ((xwhast2j=1) or (xwhast3j=1)) whast_t7=1.

If ((xwhast2k=1) or (xwhast3k=1)) whast_t8=1.

If ((xwhast2l=1) or (xwhast3l=1)) whast_t9=1.

If ((xwhast2m=1) or (xwhast2n=1) or (xwhast2o=1) or (xwhast2p=1) or (xwhast2q=1) or (xwhast2y=1) or (xwhast3m=1)

or (xwhast3n=1) or (xwhast3o=1) or (xwhast3p=1) or (xwhast3q=1) or (xwhast3y=1)) whast_t10=1.

If ((xwhast2r=1) or (xwhast3r=1)) whast_t11=1.

If ((xwhast2s=1) or (xwhast3s=1)) whast_t12=1.

If ((xwhast2t=1) or (xwhast3t=1)) whast_t13=1.

If ((xwhast2u=1) or (xwhast3u=1) or (xwhast2z=1) or (xwhast3aa=1)) whast_t14=1.

If ((xwhast2a=1) or (xwhast2b=1) or (xwhast2c=1) or (xwhast2v=1) or (xwhast2w=1) or (xwhast2x=1) or (xwhast2aa=1)

or (xwhast2bb=1) or (xwhast2cc=1) or (xwhast2dd=1) or (xwhast2gg=1) or (xwhast2jj=1) or (xwhast2kk=1) or (xwhast2ll=1)

or (xwhast2mm=1) or (xwhast2nn=1) or (xwhast2oo) or (xwhast2ii=1)

or (xwhast3a=1) or (xwhast3b=1) or (xwhast3c=1) or (xwhast3v=1) or (xwhast3w=1) or (xwhast3x=1) or (xwhast3bb=1)

or (xwhast3cc=1) or (xwhast3dd=1) or (xwhast3ee=1) or (xwhast3hh=1) or (xwhast3kk=1) or (xwhast3ll=1) or (xwhast2mm=1)

or (xwhast3nn=1) or (xwhast3oo=1) or (xwhast3pp) or (xwhast3jj=1)) whast_t15=1.

If ((xwhast2ee=1) or (xwhast2ff=1) or (xwhast2hh=1) or (xwhast3ff=1) or (xwhast3gg=1) or (xwhast3ii=1))
whast_t16=1.

Variable labels whast_t1 'Personal theft: stolen briefcase/handbag/shopping bag'

whast_t2 'Personal theft: stolen purse/wallet'

whast_t3 'Personal theft: stolen cash'

whast_t4 'Personal theft: stolen cheque book'

whast_t5 'Personal theft: stolen credit/debit/store cards'

whast_t6 'Personal theft: stolen mobile phone'

whast_t7 'Personal theft: stolen jewellery'

whast_t8 'Personal theft: stolen clothing'

whast_t9 'Personal theft: stolen documents'

whast_t10 'Personal theft: stolen computers/other electrical goods'

whast_t11 'Personal theft: stolen CDs/tapes/videos/DVDs'

whast_t12 'Personal theft: stolen house keys'

whast_t13 'Personal theft: stolen car keys'

whast_t14 'Personal theft: stolen tools or work materials'

whast_t15 'Personal theft: other stolen items'

whast_t16 'Personal theft: stolen food/toiletries/cigarettes'.

Value labels whast_t1 to whast_t16

0 'No'

1 'Yes'.

*Vehicle theft: combining categories of items stolen.

If (stolitem=1 or v71=1) vehstole=0.

If (whast2a=1 or whast2b=1 or whast3a=1 or whast3a=1) vehstole=1.

Variable labels vehstole 'What was stolen: car/van/motorcycle/motorised scooter/moped'.

Value labels vehstole

1 'Vehicle stolen'

0 'Vehicle not stolen'.

If (stolitem=1 or v71=1) valstol=0.

If (whast2d=1 or whast3d=1) valstol=1.

If (whast2e=1 or whast3e=1) valstol=1.

If (whast2f=1 or whast3f=1) valstol=1.

If (whast2g=1 or whast3g=1) valstol=1.

If (whast2h=1 or whast3h=1) valstol=1.

If (whast2j=1 or whast3j=1) valstol=1.

If (whast2k=1 or whast3k=1) valstol=1.

If (whast2l=1 or whast3l=1) valstol=1.

Variable labels valstol 'What was stolen from car: valuables'.

Value labels valstol

1 'Valuables stolen'

0 'Valuables not stolen'.

If (stolitem=1 or v71=1) electric=0.

If (whast2m=1 or whast3m=1) electric=1.

If (whast2n=1 or whast3n=1) electric=1.

If (whast2q=1 or whast3q=1) electric=1.

If (whast2y=1 or whast3y=1) electric=1.

Variable labels electric 'What was stolen from car: electrical items'.

Value labels electric

1 'Electrical items stolen'

0 'Electrical items not stolen'.

If (stolitem=1 or v71=1) hifi=0.

If (whast2o=1 or whast3o=1 or vehpar2a=1 or vehpar3a=1) hifi=1.

Variable labels hifi 'What was stolen from car: hifi/radio/speakers'.

Value labels hifi

1 'Hifi/radio/speakers stolen'

0 'Hifi/radio/speakers not stolen'.

*JH 2006/07 -In car telephone no longer a category in items stolen from a car.

If (stolitem=1 or v71=1) cartele=0.

If (whast2i=1 or whast3i=1 or vehpar2b=1) cartele=1.

Variable labels cartele 'What was stolen from car: mobile/in car telephone'.

Value labels cartele

1 'Mobile/ In car telephone stolen'

0 'Mobile/ In car telephone not stolen'.

If (stolitem=1 or v71=1) tools=0.

If (whast2u=1 or whast3u=1 or vehpar2i=1 or vehpar3g=1) tools=1.

Variable labels tools 'What was stolen from car: tools'.

Value labels tools

1 'Tools stolen'

0 'Tools not stolen'.

If (stolitem=1 or v71=1) bike=0.

If (whast2v=1 or whast3v=1) bike=1.

Variable labels bike 'What was stolen from car: bicycle'.

Value labels bike

1 'Bicycle stolen'

0 'Bicycle not stolen'.

If (stolitem=1 or v71=1) camera=0.

If (whast2p=1 or whast3p=1) camera=1.

Variable labels camera 'What was stolen from car: camera'.

Value labels camera

1 'Camera stolen'

0 'Camera not stolen'.

If (stolitem=1 or v71=1) cdtapedvd=0.

If (whast2r=1 or whast3r=1) cdtapedvd=1.

Variable labels cdtapedvd 'What was stolen from car: CDs/tapes/videos/DVDs'.

Value labels cdtapedvd

1 'CDs etc stolen'

0 'CDs etc not stolen'.

If (stolitem=1 or v71=1) hhitems=0.

If (whast2cc=1 or whast2jj=1 or whast2kk=1 or whast2ll=1 or whast3dd=1 or whast3kk=1 or whast3ll=1 or whast3mm=1) hhitems=1.

Variable labels hhitems 'What was stolen from car: household items'.

Value labels hhitems

1 'Household items stolen'

0 'Household items not stolen'.

If (stolitem=1 or v71=1) foodcigs=0.

If (whast2ee=1 or whast2ff=1 or whast2hh=1 or whast3ff=1 or whast3gg=1 or whast3ii=1) foodcigs=1.

Variable labels foodcigs 'What was stolen from car: food/cigarettes/toiletries'.

Value labels foodcigs

1 'Food/cigarettes/toiletries stolen'

0 'Food/cigarettes/toiletries not stolen'.

If (stolitem=1 or v71=1) fuel=0.

if (vehpar2k=1 or vehpar3h=1) fuel=1.

Variable labels fuel 'What was stolen from car: fuel/petrol/diesel'.

Value labels fuel

1 'Fuel stolen'

0 'Fuel not stolen'.

If (stolitem=1 or v71=1) Hkeys=0.

if (whast2s=1 or whast3s=1) Hkeys=1.

Variable labels Hkeys 'What was stolen from car: house keys'.

Value labels Hkeys

1 'House keys stolen'
0 'House keys not stolen'.

If (stolitem=1 or v71=1) Ckeys=0.
if (whast2t=1 or whast3t=1) Ckeys=1.
Variable labels Ckeys 'What was stolen from car: car keys'.
Value labels Ckeys
1 'Car keys stolen'
0 'Car keys not stolen'.

If (stolitem=1 or v71=1) tax=0.
if (vehpar2l =1) tax=1.
Variable labels tax 'What was stolen from car: tax disc'.
Value labels tax
1 'Tax disc stolen'
0 'Tax disc not stolen'.

*JH 2006/07 - Exterior fittings no longer a category in items stolen from a car, instead amalgamated equivalent categories from 2005/06.
*Categories included: hub caps, wheel trims/dust caps, aerials, number plates, makers badge, wingmirrors, windscreen wipers, lights.
*Categories not available: exhaust, luggage/bicycle carrier/rack, so note comparability.
If (stolitem=1 or v71=1) extfit=0.
if (vehpar2f=1 or vehpar3i=1 or vehpar3j=1 or vehpar3k=1 or vehpar3l=1 or vehpar3m=1 or vehpar3n=1 or vehpar3o=1 or vehpar3p=1) extfit=1.
Variable labels extfit 'What was stolen from car: exterior fittings'.
Value labels extfit
1 'Exterior fittings stolen'
0 'Exterior fittings not stolen'.

If (stolitem=1 or v71=1) tyres=0.
if (vehpar2g =1 or vehpar2h=1 or vehpar3d=1 or vehpar3e=1) tyres=1.
Variable labels tyres 'What was stolen from car: tyres/wheels'.
Value labels tyres
1 'Tyres/wheels stolen'
0 'Tyres/wheels not stolen'.

If (stolitem=1 or v71=1) othervp=0.
If (vehpar2d=1 or vehpar2e=1 or vehpar2j=1 or vehpar2m=1 or vehpar2o=1 or vehpar2n=1 or vehpar3f=1 or vehpar3c=1 or vehpar3s=1) othervp=1.
Variable labels othervp 'What was stolen from car: other vehicle parts'.
Value labels othervp
1 'Other vehicle parts stolen'
0 'Other vehicle parts not stolen'.

If (stolitem=1 or v71=1) otherst=0.
If (vehpar2c=1 or whast2oo=1 or whast2w=1 or whast2x=1 or whast2bb=1 or whast2dd=1 or whast2z=1 or whast2gg=1 or whast2mm=1 or whast2nn=1 or whast2ii=1 or whast3pp=1 or whast3w=1 or whast3x=1 or whast3cc=1 or whast3ee=1 or whast3aa=1 or whast3hh=1 or whast3nn=1 or whast3oo=1 or whast3jj=1) otherst=1.
Variable labels otherst 'What was stolen from car: other items'.
Value labels otherst
1 'Other items stolen'
0 'Other items not stolen'.

If (stolitem=1 or v71=1) garden=0.
if (whast2w=1 or whast3w=1) garden=1.
Variable labels garden 'What was stolen from car: garden furniture/ornaments/plants/equipment'.
Value labels garden
1 'Garden furniture stolen'
0 'Garden furniture not stolen'.

If (stolitem=1 or v71=1) glasses=0.

if (whast2ii=1 or whast3jj=1) glasses=1.

Variable labels glasses 'What was stolen from car: glasses/sunglasses/spectacles'.

Value labels glasses

1 'Glasses stolen'

0 'Glasses not stolen'.

*For 2006/07 removing garden furniture and glasses etc from other category.

If (stolitem=1 or v71=1) otherst2=0.

If (vehpar2c=1 or whast2oo=1 or whast2x=1 or whast2bb=1 or whast2dd=1 or whast2z=1 or whast2gg=1 or whast2mm=1 or whast2nn=1 or whast3pp=1

or whast3x=1 or whast3cc=1 or whast3ee=1 or whast3aa=1 or whast3hh=1 or whast3nn=1 or whast3oo=1) otherst2=1.

Variable labels otherst2 'What was stolen from car: other items excl garden furniture/glasses'.

Value labels otherst2

1 'Other items stolen'

0 'Other items not stolen'.

*WHETHER RESPONDENT HAD AN EMOTIONAL REACTION TO AN INCIDENT.

if (emotreac=2) respreac=0.

if (emotreac=1) respreac=1.

variable labels respreac 'Whether respondent had an emotional reaction to an incident'.

value labels respreac

0 'Respondent not emotionally affected'

1 'Respondent emotionally affected'.

*DEGREE OF EMOTIONAL IMPACT FOR RESPONDENT.

if (emotreac=2) totaff=1.

if (howaff1=1) totaff=2.

if (howaff1=2) totaff=3.

if (howaff1=3) totaff=4.

variable labels totaff 'Degree of emotional impact for respondent'.

value labels totaff

1 'Respondent not affected'

2 'Respondent affected very much'

3 'Respondent affected quite a lot'

4 'Respondent affected a little'.

*TYPE OF EMOTIONAL REACTION RESPONDENT EXPERIENCED.

if (emotreac=1 or emotreac=2) anger=0.

if (whemota = 1) anger=1.

if (emotreac=1 or emotreac=2) shock=0.

if (whemotb = 1) shock=1.

if (emotreac=1 or emotreac=2) fear=0.

if (whemotc = 1) fear=1.

if (emotreac=1 or emotreac=2) depress=0.

if (whemotd = 1) depress=1.

if (emotreac=1 or emotreac=2) anxiety=0.

if (whemote = 1) anxiety=1.

if (emotreac=1 or emotreac=2) confid=0.

if (whemotf = 1) confid=1.

if (emotreac=1 or emotreac=2) sleeping=0.

if (whemotg = 1) sleeping=1.

if (emotreac=1 or emotreac=2) crying=0.

if (whemoth = 1) crying=1.

if (emotreac=1 or emotreac=2) annoy=0.

if (whemoti = 1) annoy=1.

if (emotreac=1 or emotreac=2) othemot=0.

if (whemotj = 1) othemot=1.

if (emotreac=1 or emotreac=2) othemot2=0.

if (whemotd = 1 or whemote = 1 or whemotf = 1 or whemoti = 1 or whemotj = 1) othemot2=1.

variable labels anger 'Emotional reaction: anger'

shock 'Emotional reaction: shock'

fear 'Emotional reaction: fear'

depress 'Emotional reaction: depression'

anxiety 'Emotional reaction: anxiety/panic attacks'

confid 'Emotional reaction: loss of confidence/feeling vulnerable'

sleeping 'Emotional reaction: difficulty sleeping'

crying 'Emotional reaction: crying/tears'

annoy 'Emotional reaction: annoyance'

othemot 'Emotional reaction: other'

othemot2 'Emotional reaction: grouped other'.

value labels anger shock fear sleeping crying depress anxiety confid annoy othemot othemot2

1 'Yes - experienced that reaction'

0 'No - did not experience that reaction'.

*VICTIM SATISFACTION.

recode satpol (1 thru 4=copy) (else = sysmis) into satisfa2.

variable labels satisfa2 'Victim satisfaction (4 groups excl DK & RF)'.

value labels satisfa2

1 'Very satisfied'

2 'Fairly satisfied'

3 'A bit dissatisfied'

4 'Very dissatisfied'.

recode satpol (1,2 = 1) (3,4 = 2) (else = sysmis) into satisgp2.

variable labels satisgp2 'Victim satisfaction (2 groups)'.

value labels satisgp2

1 'Satisfied'

2 'Dissatisfied'.

*COST OF HOUSEHOLD THEFT.

recode totvalue (0=1) (1 thru 19=2) (20 thru 49=3) (50 thru 99=4) (100 thru 199=5) (200 thru 499=6) (500 thru 999=7) (1000 thru 9999=8) into totval2.

variable labels totval2 'Grouped total replacement value of what was stolen'.

value labels totval2

1 'No cost'

2 '£1 to £19'

3 '£20 to £49'

4 '£50 to £99'

5 '£100 to £199'

6 '£200 to £499'

7 '£500 to £999'

8 '£1,000 or more'.

*ANY INFORMATION ABOUT THE OFFENDER.

if (v78=1 or descroff=1) seeany=1.

if (descroff=2) seeany=2.

variable labels seeany 'Whether any information about the offender'.

value labels seeany
1 'Victim saw offender'
2 'Victim did not see offender'.

***NUMBER OF OFFENDERS - RECODED.**

recode NumOff (1 thru 4 = copy) (else=sysmis) into Numoff2.
variable labels Numoff2 'Number of offenders'.
value labels Numoff2
1 'One'
2 'Two'
3 'Three'
4 'Four or more'.

***SEX OF OFFENDER.**

if (offsex1=1 or offsex=1) ofsex=1.
if (offsex1=2 or offsex=2) ofsex=2.
if (offsex=3) ofsex=3.
variable labels ofsex 'Sex of offender(s)'.
value labels ofsex
1 'Male'
2 'Female'
3 'Both sexes'.

***AGE OF OFFENDER.**

*Re-basing multicode responses to the full number of valid incidents.
compute under=0.
if (ageoffA=1) under = 1.
compute school=0.
if (ageoffB=1) school = 1.
compute youth = 0.
if (ageoffC=1) youth = 1.
compute adult=0.
if (ageoffD=1) adult = 1.
compute older=0.
if (ageoffE=1) older = 1.

variable labels under 'Offender: under school age (one offender)'
school 'Offender: child of school age (one offender)'
youth 'Offender: aged 16-24 (one offender)'
adult 'Offender: aged 25-39 (one offender)'
older 'Offender: aged 40+ (one offender)'.
value labels under school youth adult older
1 'Yes'
0 'No'.

*Merging responses from single code (where one offender) and multicode variables.
compute ageof1=0.
if (ageoff1=1 or under=1) ageof1=1.
compute ageof2=0.
if (ageoff1=2 or school=1) ageof2=1.
compute ageof3=0.
if (ageoff1=3 or youth=1) ageof3=1.
compute ageof4=0.
if (ageoff1=4 or adult=1) ageof4=1.
compute ageof5=0.
if (ageoff1=5 or older=1) ageof5=1.

variable labels ageof1 'Offender: under school age (all offenders)'
ageof2 'Offender: child of school age (all offenders)'
ageof3 'Offender: aged 16-24 (all offenders)'

ageof4 'Offender: aged 25-39 (all offenders)'
ageof5 'Offender : aged 40+ (all offenders)'.
value labels ageof1 ageof2 ageof3 ageof4 ageof5
1 'Yes'
0 'No'.

*WHETHER A WEAPON WAS USED.

if (v78=1 or descroff=1) offinfo1=1.
if ((vftype = 1) and (descroff=2)) offinfo1=0.
if ((vftype = 2) and (v78=2)) offinfo1=0.
if (wherhapp=2 and v78=2) offinfo1=0.
variable labels offinfo1 'Information about offender'.

if (weapon = 1) weap1 = 1.
if (weapon = 2) weap1 = 2.
if any (weapon, 8,9) weap1 = 9.
if (offinfo1 = 0) weap1 = 3.
variable labels weap1 'Whether offender had weapon'.
value labels weap1
1 'Yes'
2 'No'
3 'No information about offender'
9 'Don"t know/refused'.

*Excl DK/ref from weapon DV.

if (weap1 = 1) weap2 = 1.
if (weap1 = 2) weap2 = 2.
if (weap1 = 3) weap2 = 3.
variable labels weap2 'Whether offender had weapon (excl DK/RF)'.
value labels weap2
1 'Yes'
2 'No'
3 'No information about offender'.

*INDIVIDUAL TYPES OF WEAPONS.

compute glass = 0.
if (whatweaa = 1 or whatweab = 1) glass = 1.
variable labels glass 'Whether used bottle or glass as a weapon'.

compute knife = 0.
if (whatweac = 1) knife = 1.
variable labels knife 'Whether used knife as a weapon'.

compute stabbing = 0.
if (whatwead = 1) stabbing = 1.
variable labels stabbing 'Whether used screw driver/stabbing implement as a weapon'.

compute hitting = 0.
if (whatweae = 1) hitting = 1.
variable labels hitting 'Whether used stick/club/hitting implement as a weapon'.

compute gun = 0.
if (whatweaf = 1 or whatweag = 1 or whatweah = 1 or whatweai = 1) gun = 1.
variable labels gun 'Whether used gun as a weapon'.

compute syringe = 0.
if (whatweaj = 1) syringe = 1.
variable labels syringe 'Whether used syringe as a weapon'.

compute stones = 0.

if (whatweak = 1) stones = 1.
variable labels stones 'Whether used stones/bricks/concrete as a weapon'.

compute othweap = 0.
if (whatweal = 1) othweap = 1.
variable labels othweap 'Whether used something else as a weapon'.

compute othweap2 = 0.
if (whatweaj = 1 or whatweak = 1 or whatweal = 1) othweap2 = 1.
variable labels othweap2 'Whether used syringe/stones/something else as a weapon'.

*Excl DK/RF from type of weapon DV.

if any (weapon, 8,9) glass=9.
if any (weapon, 8,9) gun=9.
if any (weapon, 8,9) knife=9.
if any (weapon, 8,9) stabbing=9.
if any (weapon, 8,9) hitting=9.
if any (weapon, 8,9) syringe=9.
if any (weapon, 8,9) stones=9.
if any (weapon, 8,9) othweap=9.
if any (weapon, 8,9) othweap2=9.

if (glass = 0) glass1 = 0.
if (glass = 1) glass1 = 1.
variable labels glass1 'Whether used bottle or glass as a weapon (excl DK/RF)'.

if (knife = 0) knife1 = 0.
if (knife = 1) knife1 = 1.
variable labels knife1 'Whether used knife as a weapon (excl DK/RF)'.

if (stabbing = 0) stab1 = 0.
if (stabbing = 1) stab1 = 1.
variable labels stab1 'Whether used screw driver/stabbing implement as a weapon (excl DK/RF)'.

if (hitting = 0) hit1 = 0.
if (hitting = 1) hit1 = 1.
variable labels hit1 'Whether used stick/club/hitting implement as a weapon (excl DK/RF)'.

if (gun = 0) gun1 = 0.
if (gun = 1) gun1 = 1.
variable labels gun1 'Whether used gun as a weapon (excl DK/RF)'.

if (syringe = 0) syr1 = 0.
if (syringe = 1) syr1 = 1.
variable labels syr1 'Whether used syringe as a weapon (excl DK/RF)'.

if (stones = 0) stones1 = 0.
if (stones = 1) stones1 = 1.
variable labels stones1 'Whether used stones/bricks/concrete as a weapon (excl DK/RF)'.

if (othweap = 0) othweap1 = 0.
if (othweap = 1) othweap1 = 1.
variable labels othweap1 'Whether used something else as a weapon (excl DK/RF)'.

if (othweap2 = 0) othweap3 = 0.
if (othweap2 = 1) othweap3 = 1.
variable labels othweap3 'Whether used syringe/stones/something else as a weapon (excl DK/RF)'.

value labels glass1 gun1 knife1 stab1 hit1 syr1 stones1 othweap1 othweap3
1'Yes'
0'No, not used'.

***RELATIONSHIP TO VICTIM.**

*knewoff and knewoff1- DK has not been recorded at code 4 or 3 as it should have been but at code 9.

*Correcting for this by recoding variable. (However seenoff and seenoff1 which route off knewoff and knewoff1 respectively do pick up the sample set they should).

```
recode knewoff (1 thru 3 = copy) (sysmis=sysmis) (else=4).  
recode knewoff1 (1 thru 2 = copy) (sysmis=sysmis) (else=3).
```

*Creating a variable that identifies those who could answer the questions 'had you seen offender before' from those who did not answer question or did not know.

```
recode seenoff (1 thru 2=1) (else=2) into sceenoff.
```

```
recode seenoff1 (1 thru 2=1) (else=2) into scenoff1.
```

variable labels seenoff 'Whether victim knew if they had seen multiple offenders before'

seenoff1 'Whether victim knew if they had seen single offender before'.

value labels seenoff seenoff1

1 'Knew whether offender had been seen before'

2 'Did not know whether offender had been seen before'.

*Single offender.

*Strangers (includes those who could not answer SeenOff).

```
if (seenoff1=2) relate2=1.
```

```
if ((knewoff1=2) and (scenoff1=2)) relate2=1.
```

```
if ((knewoff1=3) and (scenoff1=2)) relate2=1.
```

*Known casually.

```
if (howknow1=1 or howknow1=2) relate2=2.
```

*Known well.

```
if (howknow1=3) relate2=3.
```

```
if (wellknow=1) relate2=3.
```

variable labels relate2 'Single offender: relationship to victim'.

value labels relate2

1 'Strangers'

2 'Casual'

3 'Knew well'.

*Multiple offenders.

*Strangers.

```
if (seenoff=2) relate3=1.
```

```
if ((knewoff=3) and (sceenoff=2)) relate3=1.
```

```
if ((knewoff=4) and (sceenoff=2)) relate3=1.
```

*Known casually.

```
if (((howknowa=1) or (howknowb=1)) and (howknowc=0)) relate3=2.
```

*Known well.

```
if (howknowc=1) relate3=3.
```

variable labels relate3 'Multiple offender: relationship to victim'.

value labels relate3

1 'Strangers'

2 'Casual'

3 'Knew well'.

*Adding multiple offender variable into single offender variable

```
if (relate3=1) relate2=1.
```

```
if (relate3=2) relate2=2.
```

```
if (relate3=3) relate2=3.
```

*Setting up the relate variable.

```

if ((vftype=1) & any(v78,2,8,9) & any(descroff,2,8,9)) relate=1.
if ((vftype=1) & any(knewoff1,2,3,4,5,6,7,8,9) & any(seenoff1,2,3,4,5,6,7,8,9))
  relate=1.
if ((vftype=1) & any(knewoff,3,4,5,6,7,8,9) & any(seenoff,2,3,4,5,6,7,8,9)) relate=1.
if ((vftype=2) & any(v78,2,8,9)) relate=1.
if ((vftype=2) & wellknow=4) relate=1.
if ((vftype=1) & any(howknow1,1,2)) relate=2.
if ((vftype=1) & (howknowa=1 or howknowb=1)) relate=2.
if ((vftype=2) & any(wellknow,2,3)) relate=2.
if ((vftype=1) & howknow1=3) relate=3.
if ((vftype=1) & (howknowc=1)) relate=3.
if ((vftype=2) & wellknow=1) relate=3.
recode relate (sysmis=0).
variable labels relate 'How well victim knew offender'.
value labels relate
  1 'Strangers'
  2 'Casual'
  3 'Knew well'.

```

*Creating broad offence grouping variable.

```

if any (offence, 11, 12, 13, 21, 32, 33, 41, 42) offgroup=1.
if any (offence, 91, 92, 93, 94) offgroup=2.
variable labels offgroup 'Broad offence group for BCS violence (excl snatch theft)'.
value labels offgroup
  1 'All BCS violence (excl snatch theft)'
  2 'Threats'.

```

*Setting up the violgrp variable.

*Offrel variables changed in 2006/07, from offrel2, offrel2a etc to offrel3, offrel3a.

*Additional category of 'ex-partner of someone else in household' - included in acquaintance violence.

*In 2006/07 annual data there are some cases with 2005/06 questionnaire, so the syntax below combines the responses.

```

if (any(offence,11,12,13,21,32,33) & (relate=1)) violgrp=3.
if (any(offence,11,12,13,21,32,33) & any(relate,2,3)) violgrp=4.
if (any(offence,11,12,13,21,32,33) & any(offrel2,1,2,3,4,5,6,7)) violgrp=1.
if (any(offence,11,12,13,21,32,33) & any(offrel3,1,2,3,4,5,6,7)) violgrp=1.
if (any(offence,11,12,13,21,32,33) & (offrel2a=1 or offrel2b=1 or offrel2c=1 or offrel2d=1 or offrel2e=1 or
offrel2f=1 or offrel2g=1)) violgrp=1.
if (any(offence,11,12,13,21,32,33) & (offrel3a=1 or offrel3b=1 or offrel3c=1 or offrel3d=1 or offrel3e=1 or
offrel3f=1 or offrel3g=1)) violgrp=1.
if any(offence,41,42,43) violgrp=2.
variable labels violgrp 'BCS Type of violence'.
value labels violgrp
  1 'Domestic'
  2 'Mugging'
  3 'Stranger'
  4 'Acquaintance'.

```

*Setting up the 'violent' variable.

```

if any(offence,11,12,32,33) violent=1.
if any(offence,41,42) violent=2.
if any(offence,13,21) violent=3.
if any(offence,43) violent=4.
format violent (f2.0).
variable labels violent 'Type of violent offence (incl snatch theft) for trends'.
value labels violent
  1 'Wounding'

```

2 'Robbery'
3 'Common assault'
4 'Snatch theft'.

*Violent incidents involving injury/not.

compute respinj = 0.
if (injury1 = 1) respinj = 1.
variable labels respinj 'Whether respondent injured in any way'.

*2006/07 violent offences involving injury or not - excluding snatch theft.

if ((offence = 11 or offence = 12 or offence = 13 or offence = 21 or offence = 32 or offence = 33
or offence = 41 or offence = 42) and (respinj = 1)) violinj2 = 1.
variable labels violinj2 'Violence with injury (no snatch theft)'.

if ((offence = 11 or offence = 12 or offence = 13 or offence = 21 or offence = 32 or offence = 33
or offence = 41 or offence = 42) and (respinj = 0)) violnoi2 = 1.
variable labels violnoi2 'Violence without injury (no snatch theft)'.

*Creating new offence grouping 2006/07 incl two common assault categories - with and without injury.

if any(offence,11,12,32,33) violent2=1.
if any(offence,41,42) violent2=2.
if (any(offence,13,21) and respinj=1) violent2=3.
if (any(offence,13,21) and respinj=0) violent2=4.
format violent (f2.0).
variable labels violent2 'Type of violent offence (assault with & without injury, excl snatch theft)'.
value labels violent2
1 'Wounding'
2 'Robbery'
3 'Assault with injury'
4 'Assault without injury'.

*TYPES OF PHYSICAL INJURY.

compute minbruise=0.
if (whinju2a=1) minbruise=1.
Variable labels minbruise 'Minor bruising or black eye'.

compute sevbruise=0.
if (whinju2b=1) sevbruise=1.
Variable labels sevbruise 'Severe bruising'.

compute scratch=0.
if (whinju2c=1) scratch=1.
Variable labels scratch 'Scratches'.

compute cuts=0.
if (whinju2d=1) cuts=1.
Variable labels cuts 'Cuts'.

compute bones=0.
if (whinju2e=1) bones=1.
Variable labels bones 'Broken bones'.

compute nose=0.
if (whinju2f=1) nose=1.
Variable labels nose 'Broken nose'.

compute brteeth=0.
if (whinju2g=1) brteeth=1.
Variable labels brteeth 'Broken or lost teeth'.

compute chipped=0.
if (whinju2h=1) chipped=1.
Variable labels chipped 'Chipped teeth'.

compute concuss=0.
if (whinju2i=1) concuss=1.
Variable labels concuss 'Concussion or loss of consciousness'.

compute facial=0.
if (whinju2j=1) facial=1.
Variable labels facial 'Facial or head injuries (no mention of bruising)'.

compute eyeacid=0.
if (whinju2k=1) eyeacid=1.
Variable labels eyeacid 'Eye or facial injuries caused by acid, paint, sand, etc. being thrown in face'.

compute otherinj=0.
if (whinju2l=1) otherinj=1.
Variable labels otherinj 'Other injuries'.

value labels minbruis sevbruis scratch cuts bones nose brteeth chipped concuss facial eyeacid otherinj
1 'Yes - did experience'
0 'No - did not experience'.

*MEDICAL ATTENTION.

compute firstaid=0.
if (docatt3a=1) firstaid=1.
Variable labels firstaid 'Medical attention from trained first aider or St John's ambulance'.

compute para=0.
if (docatt3b=1) para=1.
Variable labels para 'Medical attention from a paramedic'.

compute nurse=0.
if (docatt3c=1) nurse=1.
Variable labels nurse 'Medical attention from a nurse'.

compute doct=0 .
if (docatt3d=1) doct=1.
Variable labels doct 'Medical attention from a doctor'.

compute dent=0.
if (docatt3e=1) dent=1.
Variable labels dent 'Medical attention from a dentist'.

compute nomeds=0.
if (docatt3f=1) nomeds=1.
Variable labels nomeds 'Mo medical attention'.

compute att=0.
if (firstaid=1 or para=1 or nurse=1 or doct=1 or dent=1) att=1.
Variable labels att 'Some form of medical attention'.

value labels firstaid para nurse doct dent nomeds att
1 'Yes - medical attention'
0 'No medical attention'.

*HOSPITAL STAY.

compute hosp2=0.
if (styhos1=1) hosp2=1.

Variable labels hosp2 'Whether needed to stay in hospital'.

value labels hosp2

1 'Yes - stayed in hospital'

0 'No - did not stay in hospital'.

*UNDER THE INFLUENCE OF DRINK/DRUGS.

*Variable for alcohol-related violence - only asked if offender school age or above.

if (drinkin1=1 or drinkinf=1) drink=1.

if (drinkin1=2 or drinkinf=2) drink=2.

if (drinkin1=3 or drinkinf=3) drink=9.

if (drinkin1=9 or drinkinf=9) drink=9.

variable labels drink 'Whether offender was under the influence of drink'.

value labels drink

1 'Yes'

2 'No'

9 'Don't know'.

if (druginf1=1 or druginf=1) drug=1.

if (druginf1=2 or druginf=2) drug=2.

if (druginf1=3 or druginf=3) drug=9.

if (druginf1=9 or druginf=9) drug=9.

variable labels drug 'Whether offender was under the influence of drugs'.

value labels drug

1 'Yes'

2 'No'

9 'Don't know'.